NO. 69 COMPUTER EXPANSION
U.S. Edition: $2.50
International Edition: $3.00

Complete Programs Included

FEB 84

MICRO™

for the Serious Computerist

COMPUTER EXPANSION

Adding Computer Senses
Using C64 Parallel Interface
Low Cost A/D Convertor
Apple Master Directory
Adding Functions to C64

0 74470 11690 02

ATARI COMPUTER OWNERS:

Pick the positively perfect, practical, peripheral package, from PERCOM DATA!

That's right... the positively perfect PERCOM DATA 5¼", floppy disk drive with a BUILT-IN PRINTER-PORT, for your Atari® 400/800 is now available!

Until now, Atari computer owners who wanted to hook a printer to their computer had only one choice... spend about $220 for an interface device. THOSE DAYS ARE OVER. PERCOM DATA has built a parallel printer-port right into its new AT88 PD model. Now you can add a quality disk drive system AND have a place to plug in a printer... WITHOUT BUYING an interface.

The AT88 S1 PD™ disk drive operates in both single density (88K bytes formatted) and double density (176K bytes formatted).

What more could you want? NO INTERFACE... a high quality PERCOM DATA disk drive... AND a built-in PRINTER-PORT... all with a price of $599.

Pick up a positively perfect PERCOM DATA disk drive, with printer-port... pronto!

For the name of an authorized PERCOM DATA Dealer near you, call our **TOLL-FREE HOTLINE 1-800-527-1222 NOW,** or write for more information.

Perfectly Priced

$599.

PERCOM DATA
CORPORATION

Expanding Your Peripheral Vision

DRIVES • NETWORKS • SOFTWARE

11220 Pagemill Road, Dallas, Texas 75243 (214) 340-5800
1-800-527-1222

Products for Commodore, Atari, Apple, and others!

NEW THE MONKEY WRENCH II
A PROGRAMMERS AID FOR ATARI 800
NEW AND IMPROVED — 18 COMMANDS
PLUGS INTO RIGHT CARTRIDGE SLOT

If you are a person who likes to monkey around with the ATARI 800, then THE MONKEY WRENCH II is for you!! Make your programming tasks easier, less time-consuming and more fun. Why spend extra hours working on a BASIC program when the MONKEY WRENCH can do it for you in seconds. It can also make backup copies of boot type cassette programs. Plugs into the right slot and works with ATARI BASIC cartridge.

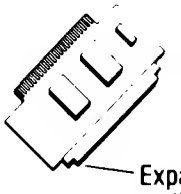
$59.95

The MONKEY WRENCH provides 18 direct mode commands. They are: AUTO LINE NUMBERING — Provides new line numbers when entering BASIC program lines. RENUMBER — Renumbers BASIC's line numbers including internal references. DELETE LINE NUMBERS — Removes a range BASIC line numbers. VARIABLES — Display all BASIC variables and their current value. Scrolling — Use the START & SELECT keys to display BASIC lines automatically. Scroll up or down BASIC program. FIND STRING — Find every occurrence of a string, XCHANGE STRING — Find every occurrence of a string and replace it with another string. MOVE LINES — Move lines from one part of program to another part of program. COPY LINES — Copy lines from one part of program to another part of program. FORMATTED LIST — Print BASIC program in special line format and automatic page numbering. DISK DIRECTORY — Display Disk Directory. CHANGE MARGINS — Provides the capability to easily change the screen margins. MEMORY TEST — Provides the capability to test RAM memory. CURSOR EXCHANGE — Allows usage of the cursor keys without holding down the CTRL key. UPPER CASE LOCK — Keeps the computer in the upper case character set. HEX CONVERSION — Converts a hexadecimal number to a decimal number. DECIMAL CONVERSION — Converts a decimal number to a hexadecimal number. MONITOR — Enter the machine language monitor.

In addition to the BASIC commands, the Monkey Wrench also contains a machine language monitor with 16 commands used to interact with the powerful features of the 6502 microprocessor.

NEW VIC RABBIT CARTRIDGE AND CBM 64 RABBIT CARTRIDGE

"High-Speed Cassette Load and Save!"



$39.95
(includes Cartridge and Manual)

Expansion Connector on the VIC Cartridge

"Don't waste your Life away waiting to LOAD and SAVE programs on Cassete Deck."

Load or Save 8K in approximately 30 seconds! Try it — your Un-Rabbitized VIC takes almost 3 minutes. It's not only Fast but VERY RELIABLE.

Almost as fast as VIC Disk Drive! Don't be foolish — Why buy the disk when you can get the VIC Rabbit for much, much less!

Easy to install — it just plugs in.
Expansion Connector on rear.
Works with or without Expansion Memory.
Works with VIC Cassette Deck.
12 Commands provide other neat features.

Also Available for 2001, 4001, and 8032

TELSTAR 64

Sophisticated Terminal Communications Cartridge for the 64.

PFO 10D 00D CP<D1>D2 BELL 12:30:00 10:14:36
(TELSTAR's Status Line)

Don't settle for less than the best!

- Upload/Download to/from disk or tape.
- Automatic File Translation.
- Communicates in Industry Standard ASCII.
- Real-Time Clock plus Alarm Clock.
- Line editing capability allows correcting and resending long command lines.
- 9 Quick Read functions.
- Menu-driven.
- Similar to our famous STCP Terminal package.
- Works with Commodore Modems and supports auto-dialing.

The best feature is the price — **only $49.95** (Cartridge and Manual)

Machine Language Monitor Cartridge

for the CBM 64

More than 20 commands allow you to access the CBM 64's Microprocessors Registers and Memory Contents. Commands include assemble, disassemble, registers, memory, transfer, compare, plus many more.

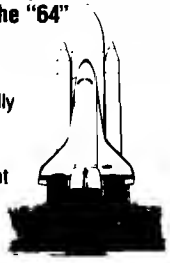
Someday every CBM 64 owner will need a monitor such as this.

Cartridge and Manual — **$24.95**

More than just an Assembler/Editor!
Now for the "64"

MAE
for
PET
APPLE
ATARI
~~$169.95~~
Only $59.95

It's a Professionally Designed Software Development System



NOW, The Best for Less!

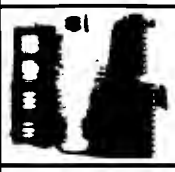
- Designed to improve Programmer Productivity
- Similar syntax and commands — No need to relearn peculiar syntaxes and commands when you go from PET to APPLE to ATARI.
- Coresident Assembler/Editor — No need to load the Editor then the Assembler then the Editor, etc.
- Also includes Word Processor, Relocating Loader and much more
- Join the ATUG User Group for MAE formatted disks
- STILL NOT CONVINCED? Send for free spec sheet!

ATARI, PET, AND CBM 64 EPROM PROGRAMMER

Programs 2716 and 2532 EPROMs. Includes hardware and software. PET = $75.00 – ATARI and CBM 64 (both include sophisticated machine language monitor) = $119.95

CBM 64 Debugger

A more sophisticated Machine Language Monitor/Debugger. 20K of object code makes this a powerful tool. Works as a symbolic debugger for the MAE assembler. Diskette and Manual — **$49.95**

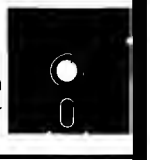


TRAP 65

TRAP 65 is a hardware device that plugs into your 6502's socket. Prevents execution of unimplemented opcodes and provides capability to extend the machines' instruction set
For PET/APPLE/SYM.
Reduced from $149.95 to $69.95

5¼ INCH SOFT SECTORED DISKETTES

Highest quality. We use them on our PETs, APPLEs, ATARIs, and other computers. $22.50/10 or $44.50/20



Prowriter Printer - Excellent dot matrix print Parallel = $489.00
Serial = $600.00 IEEE = $589.00

DC Hayes Smart Modem = $235.00
DC Hayes Micro Modem II = $289.00

Rana Disk Drive - 375
4 Drive Controller - 114

EPROMS 2716 = $4.50 2532 = $7.50
Over 40 Commodore Programs by Baker (on 4040) = $25.00

Eastern House

3239 Linda Dr.
Winston-Salem, N.C. 27106
(919) 924-2889 (919) 748-8446
Send for free catalog!

VISA
MasterCard

Editorial

Is It Reasonable?

In my December 1983 editorial, "Is There MICRO After IBM?", I stated that

> I spent five years at a company that developed one of the first "microcomputers" .... That was in 1974.

A reader wrote questioning that statement since the first true microprocessor, the 4004 was not available until about that time. He is right. What I worked on was actually a mini computer - which is why the word **microcomputer** was in quotes in my original statement. The point being made was that a 'small' computer with approximately the capabilities of the current 8-bit microcomputers could do amazing things - but that the people marketing the product could not see its extraordinary value since **IBM** had nothing like it. They could not understand what was **reasonable** for a computer system to do. They did not ask it to do enough!

My experience with this computer taught me to ask what is **reasonable** for a particular computer system to do.

Was the Apple I reasonable? I though so, enough to buy one. (This was way back in the days when a service call was answered by Steve Jobs!) It was a nice single board computer and might have had a significant impact - but was quickly made obsolete by its sibling, the Apple II.

Is the Apple II/IIe reasonable? Of course. It offered an excellent package of hardware at a low price. The fact that the company made it easy for third-party software and hardware development was also a very positive factor. (I wonder how TI would have done if they had not intentionally limited the software and hardware to their own organization?)

Was the PET reasonable? Yes, up to a point. I believe that Commodore made a serious blunder in not providing a 'real' keyboard on the earlier PET's. With a decent keyboard, I believe that PET would have given Apple II a much more difficult time.

Are the Atari, Color Computer and Commodore 64 reasonable? Yes. Each of these products are reasonably priced, have overcome early problems of reliability, documentation, support and so forth. They each have specific features that may make one more suitable for a particular user or application, but they are all good products.

Is the Coleco ADAM reasonable? Not yet. While the 'press' for this system was impressive, a simple analysis shows that a functionally equivalent system can be purchased for about the same price using components (for example, a Commodore 64 with a disk drive and dot-matrix printer). The ADAM has reliability problems and the current documentation is terrible. Will third-party software/hardware support develop? If the basic problems are solved, perhaps the ADAM will become reasonable.

Was the Apple III reasonable? I did not think so from the start. It had very limited capabilities for the price. The market apparently thought so too.

Is the Apple LISA reasonable? It has some very impressive features and capabilities. When it was first announced we considered covering it in MICRO, but decided that, at $10,000, it was not reasonable. Now that the price is being drastically reduced, with systems announced as low as $3500, it has become very reasonable, in my humble estimation, and we plan to start covering it soon.

Is the Apple MacIntosh reasonable? After the disaster of the Apple III and the initial overpricing of the LISA, it is nice to see what should be an unqualified success for Apple. It appears that they have learned from their experiences. The MAC seems to be correctly priced, well supported, and compatible with the LIZA. MICRO will definitely be covering this computer and hopes that our coverage of the 68000 over the past year or two has gotten you prepared for the new machine.

What do you think is reasonable? I would like to hear from you. What do you find reasonable in hardware, software, languages, applications, techniques, publications, and anything else relating to microcomputers.

Robert M. Tripp

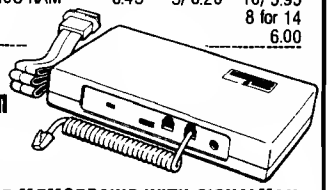
Editor-in-Chief

Hewlett Packard

Write or call for prices.

SPECIALS on INTREGATED CIRCUITS

6502	7.45	10/ 6.95	50/ 6.55	100/ 6.15
6520 PIA	5.15	10/ 4.90	50/ 4.45	100/ 4.15
6522 VIA	6.45	10/ 6.10	50/ 5.75	100/ 5.45
2716 EPROM		4.45	5/ 4.20	10/ 3.95
2532 EPROM		5.90	5/ 5.75	10/ 5.45
6116 2KX8 CMOS RAM		6.45	5/ 6.20	10/ 5.95
4116 RAM				8 for 14
4164 RAM				6.00



Anchor Automation Signalman Modems

FREE SOURCE MEMBERSHIP WITH SIGNALMAN

All Signalman Modems are Direct Connect, and provide the best price-performance values.

Dealer and OEM inquiries invited

Volksmodem with computer cable	**(99)**	**68**
Mark VI for IBM Personal Computer	(279)	139
Mark VII Auto Dial/Auto Answer	(159)	99
Mark XII Smart Model 1200/300	(399)	299
OC HAYES Smartmodem		**219**
DC Hayes Smartmodem 1200/300		529

PROM QUEEN for VIC	**170**
Apple Emulator for Commodore 64	**Call**
STAT Statistics Package for C64	**95**
Solid Oak 2 Level Stand for C64 or VIC	**29**
C64/VIC Switch (networking)	129
BACKUP V1.0 tape copier for C64 or VIC	**20**
CARDBOARD/6 Motherboard - VIC	64
CARDBOARD/5 Motherboard - C64	56
CARD PRINT G Printer Int. with Graphics	72
CARD PRINT B Printer Interface—C64/VIC	40
CARDBOARD/3s Motherboard - VIC	22
CARDCO C64/VIC Calculator Keypad	32
CARDRAM/16 RAM Expansion - VIC	44
Complete CARDCO Line in stock	
CIE and VIE IEEE Interfaces in stock	
MSD SuperDrive for C64 or IEEE	**365**
MAE Assembler for C64	50
Koala Touch Tablet—C64 or VIC	**79**
CBC 4/12 12-bit A to D Board + Software	**199**
Multiplan for C64	**79**
Dust Cover for C64 of VIC	6

APPLE—FRANKLIN ITEMS

FRANKLIN—complete line in stock	
QUENTIN Drives for Apple/Franklin	**219**
Swapper Slapper	**26**
automatic switch between paddles and joystick	
KRAFT Apple Joystick	**40**
Kraft Apple Paddle Pair	30
Koala Touch Tablet-Apple/Franklin	**90**
SPINNAKER Software in stock	
Broderbund Software in stock	
16K RAM Card for Apple	59
Multiplan—Microsoft	**185**
Solid Oak 2 Level Stand for Apple	**29**
Serial Card for Apple	89
MCP RAM/80 column card for IIe (AP/TXT)	**139**
Z80 Softcard and CP/M (Microsoft)	235
RANA Elite I with Controller	389
Parallel Printer Interface/Cable	69
Microtek and MCP interfaces in stock	
Grappler + Interface	129
Kraft Products for Apple in stock	
DC Hayes Micromodem II	299
PFS: File	**95**
PFS: Report	**95**
Videx 80 Column Card	209
Apple Blue Book	**19**

commodore

See us for Personal, Business, and Educational requirements. Educational Discounts available.

PETSCAN I $245 base price

Allows you to connect up to 30 CBM/PET Computers to shared disk drives and printers. Completely transparent to the user. Perfect for schools or multiple word processing configurations. Base configuration supports 2 computers. Additional computer hookups $100 each.

COMPACK/STCP $115

Intelligent Terminal Package for PET, CBM, C64

Includes ACIA Hardware / STCP Software

SCREENMAKER 80 Column Adapter for C64 145

Provides big screen capability for business applications.

GENESIS Computer Corp

VIController (for C64 as well) 50

combine with BSR modules for home or business control

COMSENSE Remote Sensing Adapter for C64 or VIC 35

COM VOICE Synthesizer for C64 or VIC 139

includes software for text to speech, pitch, etc.

COM CLOCK Real Time Clock with battery backup 45

VIC 20 Products and Software in stock

There EMI Software		**UMI Software**	
ABACUS Software		**HES Software**	
16K RAM for VIC	64	Vanilla Pilot	27

VICTORY Software for VIC and C64

Metamorphosis	16	Creator's Revenge	16
Labyrinth of Creator	16	Galactic Conquest	16
Kongo Kong	16	Annihilator	16
Chomper Man	16	Grave Robbers	13
Bounty Hunter	16	Adventure Pack I or II	16

PAPER CLIP Word Processor	**99**
ORACLE Data Base from Batteries Included	**109**
Super BusCard from Batteries Included	**159**
Commodore 64 Programmers Reference Guide	16
MicroChess for C64—8 levels of play	**19**
excellent graphics and color	
SPINNAKER Software C64, Apple, IBM, Atari	
Compute!'s First Book of PET/CBM	11
C64 or VIC SWITCH	**129**
POWER ROM Utilities for PET/CBM	78
WordPro 3+/64 with Spellmaster	**85**
WordPro 4+ - 8032, disk, printer	295
SPELLMASTER spelling checker for WordPro	
VISICALC for PET, ATARI, or Apple	189
PET-TRAX PET to Epson Graphics Software	**40**
SM-KIT enhanced PET/CBM ROM Utilities	**40**
Programmers Toolkit - PET ROM Utilities	35
EASY CALC for C64	**65**
PET Spacemaker II ROM Switch	36
COPYWRITER Word Processor for C64	**69**
2 Meter PET to IEEE or IEEE to IEEE Cable	40
Dust Cover for PET, CBM, 4040, or 8050	8
CmC Interfaces (ADA1800, ADA1450, SADI in stock)	
Programming the PET/CBM (Compute!) — R. West	**20**
Compute!'s Machine Language for Beginners	11
MULTIPLAN for C64	**79**
HES MODEM with Software	**65**
HES Software and Hardware in stock	
UMI products in stock	

FlexFile for PET/CBM/ C64 $ 50

Database, Report Writer with calculations, Mailing Lists.

FORTH for PET/C64 full FIG model — Cargill/Riley	**$50**
Metacompiler for FORTH for independent object code	**30**
KMMM PASCAL IV.1 for PET/CBM/C64	**95**
EARL for PET/CBM Disk-based ASSEMBLER	**65**
Super Graphics — BASIC Language Extensions	**45**
Fast machine language graphics routines for PET/CBM	
RAM/ROM for PET/CBM	4K $75 8K $90

DISK SPECIALS

Scotch (3M) 5" ss/dd	10/ 2.20	50/ 2.00	100/ 1.95
Scotch (3M) 5" ds/dd	10/ 3.05	50/ 2.80	100/ 2.75
Scotch (3M) 8" ss/sd	10/ 2.30	50/ 2.10	100/ 2.06
Scotch (3M) 8" ss/dd	10/ 2.85	50/ 2.70	100/ 2.65

We stock VERBATIM DISKS

Write for Dealer and OEM prices.

Sentinal 5" ss/dd	10/ 1.90	50/ 1.85	100/ 1.80
Sentinal 5" ds/dd	10/ 2.55	50/ 2.50	100/ 2.45

We stock Dysan disks

Wabash 5" ss/sd	10/ 1.60	50/ 1.55	100/ 1.45
Wabash 5" ss/dd	10/ 1.90	50/ 1.85	100/ 1.75
Wabash 8" ss/sd	10/ 2.00	50/ 1.95	100/ 1.85

We stock MAXELL DISKS

Write for dealer and OEM prices.

Disk Storage Pages	10 for $5	Hub Rings 50 for $6	
Disk Library Cases	8"—3.00	5"—2.25	
Head Cleaning Kits	12		

AMARAY Disk Storage Systems in stock.
Innovative Concepts FLIP 'N' FILES in stock.

CASSETTE TAPES—AGFA PE-611 PREMIUM

C-10	10/ .61	50/ .58	100/ .50
C-30	10/ .85	50/ .82	100/ .70

DATASHIELD BACKUP POWER SOURCE $265

Battery back up Uninterruptible Power Supply with surge and noise filtering. The answer to your power problems.

MultiPlan—IBM or Apple	**185**
Quadboard for IBM available	
KOALA Touch Tablets—Apple, Atari, IBM, CBM	
Peachtext 5000 Software Package	**209**
PFS Software for IBM and Apple in stock	
VOTRAX Personal Speech System	280
BMC 9191 Color Monitor	**229**
BMC 12A 12" Green Monitor	79
Brother HR-75 Daisy Wheel Printer	**769**
Dynax (Brother) DX-15 Daisy Wheel Printer	**469**
Itoh Prowriter Parallel Printer	379
Panasonic 1090 Printer with Correspondence Mode	**279**
Daisywriter 2000 with 48K buffer	1020
Gemini 10X	**299**
EPSON, Okidata, Star Micronics printers in stock	
USI CompuMOD 4 R F Modulator	29
We Stock AMDEK Monitors	
Amdek DXY-100 Plotter	590
A P Products	15% OFF
COMPUTER COVERUPS IN STOCK	
BROOKS 6 Outlet Surge Suppressor/Noise Filter	**54**
Surge Suppressor-6 outlet	29
Electrohome 1302-2 13" Hi-res RGB Monitor	335
Panasonic 12" Monitor (20 MHz) with audio	135
Synertek SYM-1 Microcomputer	**189**

ALL BOOK and SOFTWARE PRICES DISCOUNTED

USI Video Monitors—Green or AMBER 20 MHz hi-res.
Dealer and OEM inquiries invited

ZENITH data systems

ZVM-122A	109	ZVM-123G	95
ZVM-131	300	ZVM-135	510
HERO 1 Robot (factory assembled)			2145
Z29 Terminal (DEC and ADM compatible)			680
ZT-10 Intel. Terminal with Serial Port			340
Z100 16-bit/8-bit Systems			CALL

We stock entire Zenith line.

ATARI — WE STOCK ENTIRE LINE

SPINNAKER and Broderbund Software in Stock.

215-822-7727
252 Bethlehem Pike
Colmar, PA 18915

A B Computers

WRITE FOR CATALOG. Add $1.50 per order for United Parcel. We pay balance of UPS surface shipping charges on all prepaid orders (add extra for mail, APO/FPO, air). Prices include cash discount. Regular prices slightly higher. Prices subject to change.

MICRO™

for the ***Serious Computerist***

MICRO
P.O. Box 6502
Chelmsford, MA 01824
617/256-3649

Publisher/Editor-In-Chief
Robert M. Tripp

Associate Publisher
Cindy Kocher

Production Manager
Jennifer Harris

Advertising Manager
Cindy Kocher

Sales Manager
C. Skip Bentle

Circulation Manager
Linda Hensdill

Customer Service
Kim Dundas

Accounting
Donna M. Tripp

Contributing Editors
Cornelis Bongers
Phil Daley
David Malmberg
John Steiner
Jim Strasma
Paul Swanson
Richard C. Vile, Jr.
Loren Wright

MICRO is published monthly by:
MICRO, Chelmsford, MA 01824.
Second Class postage paid at:
Chelmsford, MA 01824 and additional mailing offices.
USPS Publication Number: 483470.
ISSN: 0271-9002.
Send subscriptions, change of address, USPS Form 3579, requests for back issues and all othe fulfillment questions to:
MICRO
P.O.Box 6502
Chelmsford, MA 01824
or call 617/256-3649.
Subscription Rates: (per year):
U.S. $24.00 or $42.00 for two years
Foreign surface mail: $27.00
Air mail: Europe $42.00
Mexico, Central America, Middle East, North Africa, Central Africa $48.00
South America, South Africa, Far East, Australia, New Zealand $72.00

Features

14 Using the Commodore 64 Parallel Interface
Robert M. Tripp
Programs and cable connections for driving printers and communicating with other computers

19 The Investor
Joseph Kattan
A method to help choose the best investments in today's confusing financial world.

30 Fast Low Cost A/D Convertor
Frederick J. Genett
A new chip can lessen the confusion of A/D selection

33 Adding Computer Senses to Your Micro
Andrew Cornwall
Simple hardware and software to make your computer more aware

38 DOSPLUS for Commodore 64, Part 2
Michael Keryan
More functions that can be called from the keyboard

51 Master Directory of the Apple, Part 2
Charles Hill
Sort and Print your disk directories for quick reference

Departments

2 Editorial
6 Letterbox
8 Updates and Microbes

28 From Here to Atari
Paul Swanson
My First Alphabet & Atari 1027 Printer

55 CoCo Bits
John Steiner
Assembly Language Programming

57 Interface Clinic
Ralph Tenny
More about A/D converters

61 Commodore Compass
Loren Wright
Modems and on-line information sources

63 Listing Conventions
New MICRO standardized listing conventions for Atari and Commodore computers

64 Advertiser Index

Letterbox

Dear Editors

For your readers who would like to interface their VIC-20 to a Model 33 or later Teletype, I herewith submit the instructions. The secret to this interface depends more on software than hardware. As shown on the schematic, three pins of the User Port are all that are required to access the computer output. A simple 20 ma. current loop drives the TTY. What took about two weeks to obtain was the correct program statements to send properly formatted data to the port. One cannot derive these statement from the Commodore or SAMS programming guides.

To set up the User Port requires the following statement at the head of the program:

OPEN 128,2,3,CHR$(131)

This statement must then be referenced each time printing on the TTY is desired. For instance: **PRINT# 128,"HELLO"**. The **OPEN** statement must be at the head of the program because it creates a buffer at the top of free memory. If one creates a string array before executing the **OPEN** statement, the array will be obliterated.

To **LIST** a program, the user must type the following in immediate mode:

OPEN 128,2,3,CHR$(131):
CMD 128:LIST

I hope I have saved someone the frustration I went through wondering how to make the TTY print the right stuff.

Bruce Showalter
Abilene, TX 79601

Dear Editor:

In the October 1983 issue of MICRO:65, Mr. Allen claims to show good 'structure' in his game program. If that program was handed-in in my introductory computing class at Carnegie-Mellon, it would have been graded at 70 out of 100 points.

They teach us not to declare variables as global, unless absolutely necessary. We are only allowed to declare files as global. Also, formal VAR parameters should be kept to a minimum. In other words, keep each procedure and function as **LOCAL** as possible. This makes development **EASIER**. One procedure does not feel the side effects of another.

An example may make this clear. If procedure A uses a variable **COUNT** to mean the total number of elements used in an array, and procedure B, which is called by A, uses the variable **COUNT** as the index of a FOR loop, a logic error will occur unless **COUNT** is local to procedure B. This is a common problem that BASIC programmers should understand well. The best way of avoiding such a problem is to localize as many variables as possible. Another advantage to making variables local is modularity. A procedure can be copied directly to another program without changing any variable names.

I hope I have been of some help.

Steven M. Gale
Pittsburgh, PA 15213

Dear Editor:

The following comments are directed to the article "Random Number Generator" by Bill Walker in MICRO 65:44. While 'Prime Numbers' are correctly defined it is erroneously stated that all Prime Numbers are odd and that 2 is not prime. It is later noted correctly in connection with the first example that 2 is a prime factor of 256. All primes except 2 are odd.

'Relatively Prime Numbers' are incorrectly defined. Actually two numbers are relatively prime if they have no factor other than 1 in common. For example, 39 does not divide 52 evenly but if 39 is substituted for **C** in the second example where **M** is 52 the series will repeat after 4 terms since 39 and 52 have the common factor 13 and thus are not relatively prime.

It should be noted that, unless 2 occurs more than twice or some other prime more than once in the factorization of **M**, **B** will be an integral multiple of **M** and consequently **A MOD M** will be equal to **1**, and the equation reduces to

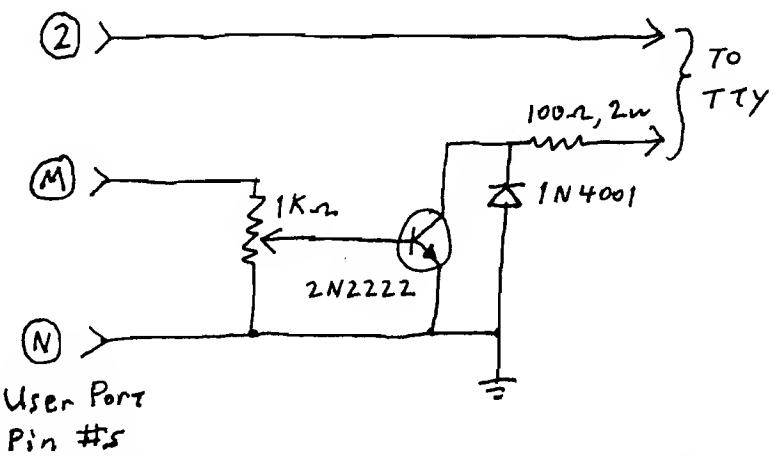
X(n + 1) = (X(n) + C) MOD M

This results in an Arithmetic Series (**MOD M**) with a common difference of **C**. Therefore, 'fiddling' with the multiplier **A** as suggested will be unfruitful and frustrating. Also, since any series generated as in example 2 will repeat after a maximum of **M** terms, changing the seed can only alter the point at which the series is entered, not the nature of the series.

Robert W. Tripp
Sarasota, FL 33581

(Ed. Note: The author of the above letter is my father, which is why I always sign everything Robert **M.** *Tripp)*

Schematic



FOR YOUR APPLE II

Industry standard products at super saver discount prices

SOFTWARE

	List	SGC
ARTSCI		
Magicalc	$149.00	$ 99.00
Magicwindow	99.00	65.95
Magicword	99.00	60.95
Special, All 3 of above	347.00	219.95
BRODERBUND		
Payroll	$395.00	$255.95
Acct/Rec.	395.00	255.95
Arcade Machine	59.95	39.95
Choplifter	34.95	24.95
Sea fox	29.95	20.95
Galaxy Wars	24.95	20.95
DATAMOST		
Property Mgr.	$295.00	$194.95
Real Estate Investment Package	179.95	118.95
Tax Beater	129.95	89.95
Zaxxon	39.95	27.95
Casino	39.95	27.95
Conquering Worlds	29.95	20.95
Mating Zone	29.95	20.95
Pandoras Box	29.95	20.95
Series Baseball	29.95	20.95
INFOROM		
Zork 1, 2, 3	$ 39.95	$ 27.95
Planet Fall	49.95	32.95
Starcross	39.95	27.95
PEACHTREE		
Peachcalc	$150.00	$ 92.95
PeachPak (G1/Ar/Ap/)	395.00	226.95
Mailing List Mgr.	250.00	151.95
SENSIBLE SOFTWARE		
Medical dictionary	$ 99.95	$ 72.95
DOS Plus	25.00	18.95
Legal Dictionary	99.95	72.95
SIR TECH SOFTWARE		
Legacy of Llylgamyn	$ 39.95	$ 29.95
Wizardry	49.96	34.95
Galactic Attack	29.95	21.95
SIRIUS		
Bandits	$ 34.95	$ 24.95
Epoch	34.95	24.95
Fly Wars	29.95	20.95
Beer Run	29.95	20.95
Outpost	29.95	20.95
Snake Byte	29.95	20.95
Sneakers	29.95	20.95
Joy Port	49.95	37.95
STRATIGIC SIMULATIONS		
Germany	$ 59.95	$ 42.95
North Atlantic 1986	59.95	42.95
Rapid Deployment Force	34.95	25.95
Road to Gettysburg	59.95	42.95
Computer Baseball	39.95	32.95
Computer Quarterback	39.95	32.95
Bomb Alley	59.95	42.95
Napoleons Campaigns	59.95	42.95

SPECIAL AND NEW

FRANKLIN COMPUTER

With 48K RAM, 1 Disk Drive, Disk Controller Card, 12" Amber Monitor, Additional 64K RAM, Magicalc, Magicword, Magicwindow, Magicspeller

Special Price **$1949.95**

"New" from Prometheus Products

ProModem® 1200 Bell 212A 300/1200 Baud Modem

A. Can be expanded into a telecommunication system as a stand-alone package.
B. Real time clock and calendar
C. Help Commands
D. Built-in diagnostics
E. Auto dialer
F. Programmable Intelligent Dialing
G. RS 232 interface
H. Internal Power Supply

OPTIONS

On line phone directory
Expandable buffer from 16K to 64K

SPECIAL ..... **$ 475.00**

BASIS 108

128K computer ..... **SPECIAL $915.00**

Also available with CPM 3.0 or DOS dual drives monitor, perfect series software or Magicalc, Magicwindow and Magicword. Call.

PROMETHEUS PRODUCTS

16K Card	**$ 60.00**
64K Expand-A-Ram	**$ 225.00**
128K Expand-A-Ram	**$ 275.00**
Graphitti Card (Specify Printer)	**$ 89.00**
P.S. 16K Buffer Card Expandable to 64K	**$ 105.00**
Applesurance Card	**$ 99.00**
VersaCard Four cards in one 1. Serial 2. Parallel 3. Real Time clock and calendar 4. BSR Port	**$ 159.95**
With Graphics	**$ 199.95**
PRT-1 Printer Card	**$ 79.00**

PRINTERS

NEC 8023	**$ 395.00**
with interface card and cables	**$ 449.00**
OKI 82A 80 col./120cps	**$ 439.00**
OKI 83A 136 col./120 cps	**$ 695.00**
Brother Letter Quality	**$ 889.00**
Transtar 315 Color Printer	**$ 469.95**

ACCESSORIES

T.G. Joystick	**$ 42.95**
Select a Port	**$ 39.95**
System Saver Fan	**$ 69.95**
Super Fan 11	**$49.95**
Slimline Disk Drives	**$ 225.00**
Disk Drives Full Ht.	**$ 195.00**
Floppy Disc Box 10	**$ 25.00**
Floppy Disc w/order	**$ 19.90**

BUSINESS SPECIAL

Includes 80 Column Z-80 Card • 16K Card • Wordstar • Supercalc

Special **$795.00**

D.O.S. Programming SPECIAL

80 column Card • Magicalc • Magicwindow • Magicspeller • 16K Card **Special** **$449.00**

3-CARD SYSTEM

80 col/16K card/Z-80 card **Special** **$319.95**

MODEMS

Smartmodem 300	$219.95
Smartmodem 1200	$499.00
D-Cat Modem	$164.00

MONITORS

14" Composite Color High Res	$260.00
Amber 12"	$129.95
Taxan RGB color	$395.00
USI Hi/Res green	$141.95

80 COLUMN CARDS

Smarterm 11	$149.95
Videx Video Term	$215.00
Ultra Term	$289.00
View Master	$139.00
Apple IIe 64K, 80 col.	$ 99.00

Z-80 CARDS

Microsoft Softcard	$229.95
Z-80 Plus	$125.00
Z-Card 11	$131.95

EDUCATIONAL SOFTWARE

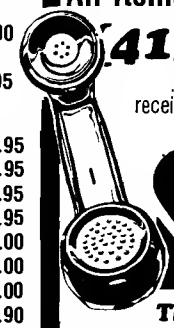
Spelling/Reading Primer	$ 29.95
Perception	$ 20.95
Counting Bee	$ 23.95
Word Scrambler	$ 15.95
Fractions	$ 36.95
Intro. Algebra	$ 21.95

BUSINESS SOFTWARE

Wordstar	$244.95
Spellstar	$159.95
WStar/MMerge	$395.00
WStar/SStar	$422.00
Supercal	$129.95
PSF File	$ 88.75
PSF Graph	$ 88.75
PSF Report	$ 88.75
PSF 3PAK	$246.75

All equipment shipped factory fresh. Manufacturers warranties included. California customers add 6½% tax. Include payment by personal check, money order, or cashier's check with order and SGC will pay shipping charge. Call for amount of shipping charge when paying by credit card.

All items are normally in stock



(415) 490-3420

... And we'll be here to help after you receive your order. Feel free to call the SGC Technical Staff for assistance.

SGC

The mail order specialists

342 Quartz Circle, Livermore, CA 94550

MICRO™

Updates and Microbes

December Error

It has been brought to our attention by reader Jeff Wisnia that the listing in Apple Cat Sort by Mark Harris (67:41) has the same error in two places. The instructions at 42EO and 42EB (lines 0145 and 0150) should read BD 00 40, and not BD 00 80.

Lots of Typos

Stephen Childress, author of OS-9: A New Operating System for the Color Computer (66:48) has caught us with our proofreading down. When the OS-9 became a reality, we rushed this article into print without giving the author time to check it. Unfortunately, our typesetter doesn't make certain characters, hence the omissions. Our apology to author and readers.Please note the following changes.

```
  Page 49, col. 3 --
OS9: list filename ))/p& should
read
OS9: list filename )/p&
  and a little further down, the
correct reading is
OS9: asm filename l o = progname
)/p)) oopsfile& edit somefile
  Page 50, col. 1 -- the correct
readings are:
OS9: fextext(filea-lf +uc + ll =
80)fileb
OS9: wordcount(filea
OS9: fixtext +ll=l(filea !
wordcount)result
report ( company/invoices + from =
      1/82 to = 12/82 ) workfile1!
      sort ( workfile1 + key = date
      ) workfile2
list workfile2 ! specialformat )
/p&
  Page 50, col. 3 -- table 1 should
read figure 2.
  figure 2 -- the second 'dsave
should be omitted
  Page 52, col. 3 -- the correct
readings are:
chx  /h0/cmds
setime(/t2)/t2
shell(/t2)/t2))/t2&
  Page 53, col. 1 -- should read
OS9: /h0/afile
          col. 2 -- should read
PRINT #PATHNUM, "HELLO"
          col. 3 -- should read
(/t2)/t2))/t2&
  Page 53, Inset 1 should read:
OS9: list/modem1)bfile
OS9: list bfile)/modem1
OS9: list bfile !
aprotocol(/modem1)/modem1
OS9: dir /d0/jim)/modem1
```

A Clock Interrupt for Your Apple

MICRO 62:36

The 'Clock Calibrator' had a minor (but fatal) bug. Line 430 which read:

430 EC =DC*86400+HC*3600-+MC*60+S-D0*84600-H0*3600-M0*60-S0

should have read

430 EC =DC*86400+HC*3600-+MC*60+**SC**-D0*84600-H0*3600-M0*60-S0

The difference is in the variable **SC** that was erroneously listed as **S**.

Listing Problems

In the November and December issues we tried a new form of listing programs for the Commodore computers. This did not turn out to be very successful. There have been numerous telephone calls about the Screen Editor programs (MICRO 66 November 1983) and the MicroCalc programs (MICRO 67 December 1983), and I expect to get some on the EPSON FX80 (MICRO 68 January 1984) listing as well. So, we are going to try for a better method.

The problem arises from the fact that the Commodore and Atari computers use special non-ASCII characters to represent important commands, colors, characters, and so forth. It is not always possible in a printed listing to differentiate between the special graphic characters, even with printers specifically manufactured for these computers. For example, a thin vertical line in a solid black block can represent several quite different functions on a Commodore. Rather than try to represent the special graphics of these computers, we are going to print text that describes the function. For example, the **HOME** function on the Commodore 64/VIC 20 that moves the cursor to the top left corner of the display is represented graphically as a reversed letter **S**. Even when this reversed **S** is 'readable' from the listing, it does not represent what is happening. Our listings will replace the single reversed **S** with string **{HOME}**. The reversed heart that represents the clear screen function will be listed as **{CLEAR}**. A complete list of these 'listing conventions' will be included in each issue of MICRO to make the translations easy. Lists for both the Commodore and the Atari are found on page 63 of this issue.

Listings of the lines that had special graphic characters in the MicroCalc for the C64 and the EPSON FX80 character generator are presented below in the new format. I hope that this new technique will make the programs presented in MICRO easier to use and understand.

```
All lines with special characters in the January 1984
issue of MICRO for the article
 "Generating Characters for the EPSON FX80 on
  a Commodore 64".
Listing started on page 34.

10 PRINT"{CLEAR}GENERATING CHARACTERS"
1010 BY=FX+13*CH:PRINT"{HOME}";
1020 FOR J=7 TO 0 STEP -1:K=2{UP ARROW}J
1440 FOR J=7 TO 0 STEP -1:K=2{UP ARROW}J
2015 PRINT"{HOME,RIGHT8}CHAR #";CH;
     "{LEFT}  {SPACE,HOME}";
2040 IF T$="{POUND}" THEN CP=225: GOTO 2025
2050 IF T$="{UP ARROW}" THEN CP=87: GOTO 2025
3000 PRINT"{HOME}";QQ$;LL$;"{RIGHT}";:INPUT CH
3020 PRINT"{HOME}";QQ$;LL$;S9$
4000 PRINT"{CLEAR}";QQ$;
     "+ ON / - OFF / {POUND} HALF / {UP ARROW} FULL"
4010 PRINT"{DOWN}F1 READ  CHAR #"
5130 PD=ASC("."):PRINT"{CLEAR,BLACK}";
5160 CH=0:F1$="{F1}":F2$="{F2}":F3$="{F3}"
5165 F4$="{F4}":F5$="{F5}":F6$="{F6}":F7$="{F7}"
5170 CL$="{RIGHT}":CR$="{LEFT}":CU$="{UP}"
5175 CD$="{DOWN}":HM$="{HOME}":CS$="{CLEAR}"
5180 ZL$=RT$+"{UP,RIGHT14}"
5190 ZR$="{LEFT17}"
5196 QQ$="{DOWN10}"
5197 LL$="{RIGHT18}"
6010 PRINT"{HOME,DOWN17}";
6210 PRINT"{HOME,DOWN16}";
6300 PRINT"{UP}";EL$;B$;" LOADED";:GOTO 2010
6600 PRINT RT$;"{UP,RIGHT19}";
6604 PRINT "{LEFT18}";
7200 PRINT QQ$;"{DOWN}";
7210 PRINT RT$;"{RIGHT16}";
7220 PRINT CH;"{LEFT}        {LEFT7}";
```

```
Update for the MicroCalc Programs from the December 1983 isssue of MICRO.

Atari Version: Change 32110 DATA with strange characters to:

32110 DATA 104,104,133,204,104,133,203,104,133,206,104,133,205,104,133,208
32111 DATA 104,133,207,160,0,177,203,145,205,230,203,208,2,230,204,230,205
32112 DATA 208,2,230,206,198,207,165,207,201,255,208,2,198,208,165,207
32113 DATA 5,208,208,224,96

This machine level program scrolls the entire screen up one line, without
disturbing the top line.
```

MicroCalc for the Commodore

Listing starts on page 24 of MICRO #67, December 1984.

```
10 PRINT"{CLEAR}":POKE53281,0:POKE53280,0:GOSUB8000
20 Q$=CHR$(34):CR$=CHR$(13):DL$=CHR$(20):RB$="{RVS,=@,RVSOFF}"
25 BL$="{=@28}":DI$="{YELLOW,^Z,RVSOFF,LEFT}":CC$="{^Q,LEFT}"
40 LL=1:GOSUB8490:PRINT"{HOME,DOWN}";:GOTO110
100 LL=1:GOSUB8500:PRINT"{HOME,DOWN}";:GOSUB9000
110 S$=S$(LL):IFRIGHT$(S$,1)="?"THENPRINT"{RVS}"BL$CR$"{UP}"TAB(10);
115 PRINTTAB(10)"{YELLOW}"S$DI$;
130 IFT$="{F7}"THEN300
135 IFT$="{F8}"THENGOSUB9000:LL=1:PRINT"{HOME,DOWN}";:GOTO110
140 IFT$="{CLEAR}"THEN100
150 IFT$=CR$ORT$="{DOWN}"THEN210
160 IFT$="{UP}"THEN270
180 IFT$="{BACK ARROW}"THENS$(LL)=S$:GOSUB5000:GOSUB9000:LL=1:GOTO110
185 IFT$="{POUND}"THENPRINT"{RVS,=@,RVSOFF}";S$(LL)=S$:GOTO1000
220 LL=LL+1:IFLL=NL+1THENLL=1:PRINTRB$;:PRINT"{HOME,DOWN}"TAB(10);:GOTO110
250 PRINTRB$"{LEFT2}"DI$;
275 IFLL=0THENLL=NL:PRINTRB$"{HOME,RIGHT10,DOWN20}";:GOTO110
280 PRINTRB$CR$"{UP2}"TAB(10);:GOTO110
300 PRINTRB$:S$(LL)=S$:GOSUB3000:GOSUB7000:
    PRINT"{HOME,DOWN}"TAB(10);:LL=1:GOTO110
1000 PRINT"{HOME,DOWN,LT GREEN}";:LL=1
1030 IFT$="{POUND}"THENPRINT"{=@,HOME,DOWN}";:C$(LL)=C$:LL=1:GOTO110
1040 IFT$=CR$ORT$="{DOWN}"THEN1100
1050 IFT$="{UP}"THEN1200
1110 LL=LL+1:IFLL=NL+1THENLL=1:PRINT"{=@,LEFT,HOME,DOWN}";:GOTO1010
1120 PRINT"{=@,LEFT}"CR$;:GOTO1010
1210 LL=LL-1:IFLL=0THENLL=NL:PRINT"{=@,HOME,DOWN20}";:GOTO1010
1220 PRINT"{=@,LEFT}"CR$"{UP2}";:GOTO1010
1310 PRINT"{=@,LEFT2}"CC$;
2010 IFT$="{F7}"ORT$=CR$ORT$="{DOWN}"ORT$="{UP}"ORT$="{BACK
     ARROW}"ORT$=DL$ORT$="{CLEAR}"ORT$="{POUND}"THENRETURN
2015 IFT$="{F8}"THENRETURN
2050 IFT$="{UP ARROW}"THEN2070
3000 PRINT"{HOME,PURPLE}CALCULATING"
3030 NEXT:PRINT"{HOME}            {YELLOW}":RETURN
5000 PRINT"{CLEAR,RVS}L{RVSOFF}OAD OR {RVS}S{RVSOFF}AVE"
5045 PRINT"{DOWN,RVS}D{RVSOFF}ISK OR {RVS}T{RVSOFF}APE":GOSUB4500
5048 INPUT"{DOWN2}NAME";NA$
5080 CLOSE1:CLOSE15:GOSUB8000:GOSUB8510:PRINT"{HOME,DOWN}";:RETURN
5090 A$="":FORII=1TONL:S$=S$(II):IFS$=""THENS$="{=B}"
5115 IFA$="{=B}"THENA$=""
5205 A$="":FORII=1TONL:S$=C$(II):IFS$=""THENS$="{=B}"
5320 IFA$="{=B}"THENA$=""
5920 PRINT"{CLEAR}"D1$" "D2$" "D3$" "D4$
7000 PRINT"{HOME,DOWN}";:FORII=1TONL:S$=S$(II):SS=S(II)
7010 X$="":IFRIGHT$(S$,1)="?"THENX$=STR$(SS)+"{RVS}"+LEFT$(BL$,24-
     LEN(STR$(SS)))
8510 PRINT"{CLEAR,DOWN}";:FORII=1TONL:S$=S$(II):C$=C$(II)
8520 PRINT"{LT GREEN}"C$LEFT$(BL$,10-LEN(C$))"{YELLOW}"S$"{RVS}"
     LEFT$(BL$,28-LEN(S$))
8530 NEXT:PRINT"{DOWN,PURPLE}"MID$(NA$,4)"{LEFT4}    ":RETURN
9000 PRINT"{HOME,PURPLE}CLEAR"
9030 PRINT"{HOME}     {YELLOW,HOME}":RETURN
9900 DATAA=8000,M=48,I=11.9,I=I/1200,D=(1-(1+I){UP ARROW}-M)/I
```

[See page 63 for Listing Conventions]

FLOPPY DISKS SALE *$1.19 ea.

Economy Model or Cadillac Quality

LORAN CERTIFIED PERSONAL COMPUTER DISK ***We have the lowest prices!*** LORAN CERTIFIED PERSONAL COMPUTER DISK

***ECONOMY DISKS**

Good quality 5¼" single sided single density with hub rings.

Bulk Pac	100 Qty.	$1.19 ea.	Total Price	$119.00
	10 Qty.	1.39 ea.	Total Price	13.90

CADILLAC QUALITY

• Each disk certified • Free replacement lifetime warranty • Automatic dust remover

For those who want cadillac quality we have the Loran Floppy Disk. Used by professionals because they can rely on Loran Disks to store important data and programs without fear of loss! Each Loran disk is 100% certified (an exclusive process) plus each disk carries an exclusive FREE REPLACEMENT LIFETIME WARRANTY. With Loran disks you can have the peace of mind without the frustration of program loss after hours spent in program development.

100% CERTIFICATION TEST

Some floppy disk manufacturers only sample test on a batch basis the disks they sell, and then claim they are certified. Each Loran disk is individually checked so you will never experience data or program loss during your lifetime!

FREE REPLACEMENT LIFETIME WARRANTY

We are so sure of Loran Disks that we give you a free replacement warranty against failure to perform due to faulty materials or workmanship for as long as you own your Loran disk.

AUTOMATIC DUST REMOVER

Just like a record needle, disk drive heads must travel hundreds of miles over disk surfaces. Unlike other floppy disks the Loran smooth surface finish saves disk drive head wear during the life of the disk. (A rough surface will grind your disk drive head like sandpaper). The lint free automatic CLEANING LINER makes sure the disk-killers (dust & dirt) are being constantly cleaned while the disk is being operated. PLUS the Loran Disk has the highest probability rate of any other disk in the industry for storing and retaining data without loss for the life of the disk.

Loran is definitely the Cadillac disk in the world

Just to prove it even further, we are offering these super LOW INTRODUCTORY PRICES

List $4.99 ea. INTRODUCTORY SALE PRICE $2.99 ea. (Box of 10 only) Total price $29.90

$3.33 ea. (3 quantity) Total price $9.99

All disks come with hub rings and sleeves in an attractive package.

DISK DRIVE CLEANER $19.95

Everyone needs a disk drive doctor

FACTS

- 60% of all drive downtime is directly related to poorly maintained drives.
- Drives should be cleaned each week regardless of use.
- Drives are sensitive to smoke, dust and all micro particles.
- Systematic operator performed maintenance is the best way of ensuring error free use of your computer system.

The Cheetah disk drive cleaner can be used with single or double sided 5¼" disk drives. The Cheetah is an easy to use fast method of maintaining efficient floppy diskette drive operation.

The Cheetah cleaner comes with 2 disks and is packed in a protective plastic folder to prevent contamination.

List $29.95 / Sale $19.95

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. Visa - MasterCard - C.O.D.

PROTECTO ENTERPRIZES (WE LOVE OUR CUSTOMERS)

BOX 550, BARRINGTON, ILLINOIS 60010

Phone 312/382-5244 to order

SANYO MONITOR SALE!!

9" Data Monitor

- 80 Columns × 24 lines
- Green text display
- East to read - no eye strain
- Up front brightness control
- High resolution graphics
- Quick start - no preheating
- Regulated power supply
- Attractive metal cabinet
- UL and FCC approved

- *15 Day Free Trial - 90 Day Immediate Replacement Warranty*

9" Screen - Green Text Display	**$ 79.00**
12" Screen - Green Text Display (anti-reflective screen)	**$ 99.00**
12" Screen - Amber Text Display (anti-reflective screen)	**$119.00**
14" Screen - Color Monitor (national brand)	**$249.00**

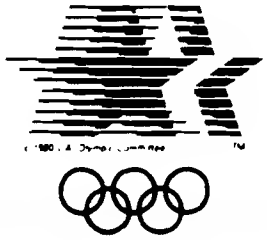
Display Monitors From Sanyo

With the need for computing power growing every day, Sanyo has stepped in to meet the demand with a whole new line of low cost, high quality data monitors. Designed for commercial and personal computer use. All models come with an array of features, including up-front brightness and contrast controls. The capacity 5 × 7 dot characters as the input is 24 lines of characters with up to 80 characters per line.

Equally important, all are built with Sanyo's commitment to technological excellence. In the world of Audio/Video, Sanyo is synonymous with reliability and performance. And Sanyo quality is reflected in our reputation. Unlike some suppliers, Sanyo designs, manufactures and tests virtually all the parts that go into our products, from cameras to stereos. That's an assurance not everybody can give you!

SANYO

Official Video Products of the Los Angeles 1984 Olympics

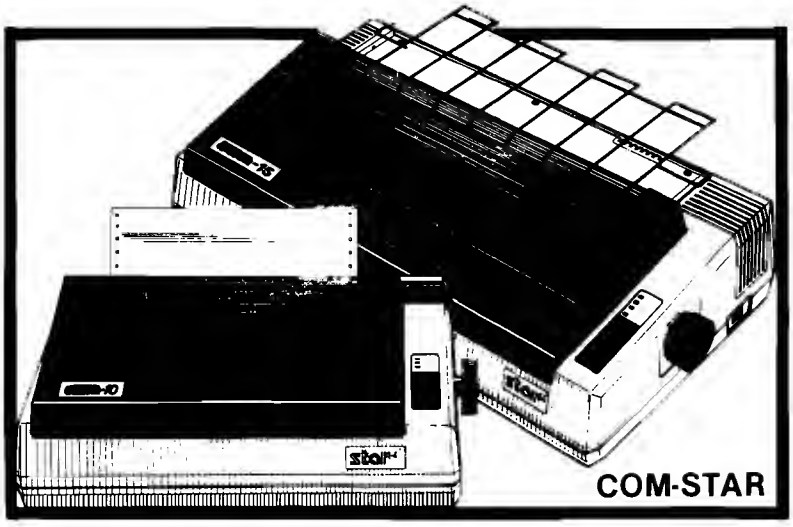


• LOWEST PRICES • 15 DAY FREE TRIAL • 90 DAY FREE REPLACEMENT WARRANTY
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. Visa - MasterCard - C.O.D.

PROTECTO
ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order



FANTASTIC PRINTER SALE

as low as **$149.00**

- *15 Day Free Trial - 180 Day Immediate Replacement Warranty*

	LIST	SALE
80 COLUMN THERMAL PRINTER — 60 CPS Bi-directional, dot matrix, prints 8½" letter size paper, full 80 columns, high resolution graphics, dot bit addressable, special symbols and true decenders! (Centronics parallel interface)	$199	**$149**
80 COLUMN TRACTOR-FRICTION PRINTER — 80 CPS Bi-directional, dot matrix, impact, prints single sheets, continuous feed paper, adjustable columns, 40 to 132 characters! Roll paper adapter $32.95. (Serial or Centronics parallel interface)	$399	**$209**
PREMIUM QUALITY 10" CARRIAGE T/F PRINTER — 120 CPS Bi-directional, impact, 9 x 9 dot matrix with double strike for 18 x 18 dot matrix. High resolution bit image (120 x 144 dot matrix) underlining back spacing, left and right margin settings, true lower decenders, with super and sub scripts. Prints standard, italic, block graphics, special characters, plus 24 of user definable characters and much more!! Prints single sheets, continuous feed and roll paper! (Centronics parallel interface)	$499	**$289**
PREMIUM QUALITY 15½" CARRIAGE PRINTER — 120 CPS Has all the features of the Premium Quality 10" Carriage T/F Printer above plus a 15½" carriage and more powerful electronic components to handle large business forms! (Centronics parallel interface)	$599	**$379**
HIGH SPEED PREMIUM QUALITY T/F 10" PRINTER — 160 CPS Save printing time with these plus features: 160 CPS speed, 100% duty cycle, 8K buffer diverse character fonts special symbols and true decenders, vertical and horizontal tabs. This is Red Hot Efficiency!!! (Serial or Centronics parallel interface)	$699	**$499**
HIGH SPEED PREMIUM QUALITY T/F 15½" PRINTER — 160 CPS Has all the features of the 10" Carriage high speed printer plus a 15½" carriage and more powerful electronics to handle larger business forms! (Serial or Centronics parallel interface)	$799	**$599**

PARALLEL PRINTER INTERFACES: (IN STOCK)

- For VIC-20 and COMMODORE 64 **$49.00**
- For all APPLE COMPUTERS **$69.00**
- For ATARI 400 and 800 COMPUTERS **$79.00**

NOTE: *Other printer interfaces are available at computer stores!*

SALE SALE

WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. We accept Visa and MasterCard. We ship C.O.D.

PROTECTO ENTERPRIZES (WE LOVE OUR CUSTOMERS)

BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Using the Commodore 64 Parallel Interface

The built-in parallel interface can be used to drive printers and communicate with other computers using the software and cable connections provided.

by Robert M. Tripp

Requires: Commodore 64

In last month's issue (MICRO 68:33), the article "Generating Characters for the EPSON FX80 on a Commodore 64" provided a programmable character program that allowed the user to define characters on the C64 display. Many printers, including the FX80, can be driven directly from the C64 *via* the Parallel Interface. Using this interface has one major advantage over the Serial Port: $$$$. Most printers come equipped with the Centronics-type parallel interface as standard. The serial port (or RS-232 port) is often an extra cost option. Who needs it! The drawback for the C64 user has been that, while the C64 has a parallel interface port capable of driving the Centronic-type printer, it has been poorly documented and no support is offered in the system software. You can not simply **OPEN** the parallel port.

Figure 1 shows the wiring list for a standard Centronic-type parallel interface for C64 (and a note explaining how to adapt it for the VIC 20). An easy-to-find (try Radio Shack) dual 12 pin edge connector plugs into the C64 end of the cable. A standard 36 pin Centronic-type parallel connector is used on the other end. If you are going to go computer-to-computer, then use whatever is required at the other computer end.

Parallel Driver Program

A program to support this parallel I/O is provided in a BASIC loader form (Listing 2) and as a annotated assembly

Figure 1

C64 Parallel I/O Connections

--- C64 ---		- Printer -		-Computer-	
Pin	Signal	Pin	Signal	Pin	Signal
A	GND	14	GND	14	GND
B	FLAG 2	10	ACK-	1	STB-
C	PB0	2	DB0	2	DB0
D	PB1	3	DB1	3	DB1
E	PB2	4	DB2	4	DB2
F	PB3	5	DB3	5	DB3
H	PB4	6	DB4	6	DB4
J	PB5	7	DB5	7	DB5
K	PB6	8	DB6	8	DB6
L	PB7	9	DB7	9	DB7
M	PA2	NC		NC	
N	GND	16	GND	16	GND
8	PC2	1	STB-	10	ACK-

Note: The connections are identical for the VIC 20 with one exception: Pin M is used in place of Pin 8, so that the STB- for the VIC to Printer or the ACK- for the VIC to Computer is on Pin M. Pin 8 is not connected.

Listing 1

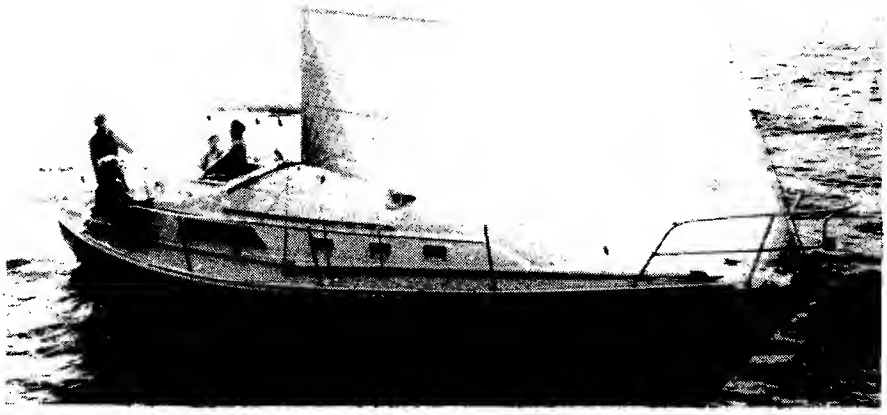
```
                        *  PARALLEL SERVICE for COMMODORE 64

                        *  by Robert M. Tripp

                        *  EQUATES

                        NXD      EQU   $00C6     KBRD CHAR COUNT
                        CHRIN    EQU   $0324     KERNAL CHAR INPUT
                        CHROUT   EQU   $0326     KERNAL CHAR OUTPUT
                        UPPLOW   EQU   $D018     UPPER/LOWER CASE REG.
                        PARA     EQU   $DD00     PARALLEL I/O REGISTER

C000 4C 0C C0   POINIT  JMP   INITPO   VECTOR TO INIT OUTPUT
C003 4C 28 C0   POSTOP  JMP   STOPPO   VECTOR TO STOP OUTPUT
C006 4C 6C C0   PIINIT  JMP   INITPI   VECTOR TO INIT INPUT
C009 4C 88 C0   PISTOP  JMP   STOPPI   VECTOR TO STOP INPUT

C00C AD 26 03   INITPO  LDA   CHROUT   COPY NORMAL OUTPUT
C00F 8D 5D C0           STA   C64OUT+1 VECTOR FOR USE BY
C012 AD 27 03           LDA   CHROUT+1 ROUTINE AFTER PARALLEL
C015 8D 5E C0           STA   C64OUT+2 OUTPUT
C018 A9 35              LDA   #OUTPUT  SETUP VECTOR TO PARALLEL
C01A 8D 26 03           STA   CHROUT   OUTPUT ROUTINE IN PLACE
C01D A9 C0              LDA   #OUTPUT/ OF NORMAL OUTPUT
C01F 8D 27 03           STA   CHROUT+1
C022 A9 FF              LDA   #$FF     SETUP PARALLEL DEVICE
C024 8D 03 DD           STA   PARA+3   FOR OUTPUT
C027 60                 RTS            DONE WITH OUTPUT INIT

C028 AD 5D C0   STOPPO  LDA   C64OUT+1 RESTORE ORIGINAL OUTPUT
C02B 8D 26 03           STA   CHROUT   VECTOR SO THAT CONTROL
C02E AD 5E C0           LDA   C64OUT+2 DOES NOT COME THROUGH
C031 8D 27 03           STA   CHROUT+1 PARALLEL SERVICE ANY
C034 60                 RTS            MORE

C035 48         OUTPUT  PHA            SAVE CHARACTER
C036 A9 02              LDA   #$02     SET TO TEST UPPER OR
C038 2C 18 D0           BIT   UPPLOW   LOWER CASE CHARACTER
C03B F0 1B              BEQ   UPPER    BRANCH ON UPPER CASE
C03D 68                 PLA            RESTORE CHARACTER
C03E 30 0D              BMI   BIT80    IS BIT 80 SET?
C040 C9 40              CMP   #$40     TEST RANGE
C042 30 13              BMI   OKAY     BELOW LOWER CASE
C044 C9 5F              CMP   #$5F
C046 10 0F              BPL   OKAY     ABOVE LOWER CASE
C048 18                 CLC            CONVERT LOWER CASE
C049 69 20              ADC   #$20     TO ASCII LOWER CASE
C04B D0 0A              BNE   OKAY     BY ADDING HEX 20

C04D C9 C0      BIT80   CMP   #$C0     TEST HIGH GRAPHICS
C04F 30 06              BMI   OKAY     IN RANGE $C0 TO
C051 C9 DF              CMP   #$DF     $DF
C053 10 02              BPL   OKAY     OKAY IF NOT IN RANGE
C055 29 7F              AND   #$7F     ELSE, MASK OFF BIT 80

C057 48         OKAY    PHA            SAVE CONVERTED CHARACTER
```

(continued)

```
C058 68        UPPER   PLA             RESTORE CHARACTER
C059 20 5F C0          JSR   PAROUT    PARALLEL OUTPUT
C05C 4C 00 00  C64OUT  JMP   $0000     ADDRESS FILLED IN BY INIT

C05F 48        PAROUT  PHA             SAVE CHARACTER
C060 A9 10             LDA   #$10      TEST PARALLEL OUTPUT READY
C062 2C 0D DD  POWAIT  BIT   PARA+$D   6526 INTERRUPT FLAG REGISTER
C065 F0 FB             BEQ   POWAIT    WAIT UNTIL OUTPUT READY
C067 68                PLA
C068 8D 01 DD          STA   PARA+1    6522 DATA REGISTER TO OUTPUT
C06B 60                RTS             RETURN

C06C AD 24 03  INITPI  LDA   CHRIN     COPY NORMAL INPUT VECTOR
C06F 8D 95 C0          STA   C64IN+1   FOR INTERNAL USE
C072 AD 25 03          LDA   CHRIN+1   TWO BYTE ADDRESS
C075 8D 96 C0          STA   C64IN+2
C078 A9 97             LDA   #PARIN    REPLACE WITH VECTOR TO
C07A 8D 24 03          STA   CHRIN     THIS PARALLEL INPUT SERVICE
C07D A9 C0             LDA   #PARIN/   TWO BYTES
C07F 8D 25 03          STA   CHRIN+1
C082 A9 00             LDA   #$00      SET PARALLEL DEVICE FOR INPUT
C084 8D 03 DD          STA   PARA+3    DIRECTION REGISTER
C087 60                RTS             THAT'S ALL TO INIT INPUT

C088 AD 95 C0  STOPPI  LDA   C64IN+1   RESTORE ORIGINAL INPUT
C08B 8D 24 03          STA   CHRIN     VECTOR TO REMOVE
C08E AD 96 C0          LDA   C64IN+2   THIS PARALLEL INPUT
C091 8D 25 03          STA   CHRIN+1   SERVICE
C094 4C 00 10  C64IN   JMP   $0000     VECTOR SET BY INITPI

C097 AD C6 00  PARIN   LDA   NXD       IF THERE IS ANY REAL C64
C09A D0 EC             BNE   STOPPI    KEYBOARD INPUT, THEN QUIT
C09C A9 10             LDA   #$10      TEST DATA PRESENT ON THE
C09E 2C 0D DD          BIT   PARA+$D   PARALLEL INPUT PORT
C0A1 F0 F4             BEQ   PARIN     WAIT FOR SOME INPUT
C0A3 AD 01 DD          LDA   PARA+1    READ PARALLEL INPUT
C0A6 48                PHA             SAVE ON STACK
C0A7 A9 02             LDA   #$02      TEST UPPER/LOWER CASE MODE
C0A9 2C 18 D0          BIT   UPPLOW    C64 STATUS REGISTER
C0AC D0 07             BNE   LOWER     LOWER CASE SET
C0AE 68                PLA             UPPER CASE SET
C0AF C9 60             CMP   #$60      TEST LOWER CASE CHAR
C0B1 30 15             BMI   PIDONE    NOT LOWER CASE
C0B3 10 0D             BPL   HIGH      LOWER CASE OR GRAPHIC

C0B5 68        LOWER   PLA             LOWER CASE SET
C0B6 C9 40             CMP   #$40      TEST UPPER CASE INPUT
C0B8 30 0E             BMI   PIDONE    NO, TOO LOW
C0BA C9 5F             CMP   #$5F      MAYBE
C0BC 10 04             BPL   HIGH      NO, TOO HIGH
C0BE 09 20             ORA   #$20      CONVERT TO DISPLAY UPPER CASE
C0C0 D0 06             BNE   PIDONE    BRANCH ALWAYS

C0C2 C9 80     HIGH    CMP   #$80      TEST BIT 80 ON
C0C4 10 02             BPL   PIDONE    LEAVE GRAPHICS ALONE
C0C6 29 5F             AND   #$5F      CONVERT LOWER TO UPPER CASE
C0C4 6C 26 03  PIDONE  JMPI  CHROUT    DISPLAY CHAR AND RETURN
```

JACK LEVERENZ CRUISING SCHOOL

Learn to Cruise in Florida



O'Day 34's (above) are used in 7-day Basic Course. O'Day 34's and 39's are used for Advanced Courses. All boats are less than two years old.

Combine the best ingredients of professional cruising instruction and a dream vacation! Here are four cruising programs designed for each level of sailing experience from beginning sailor to advanced cruiser. The programs are also perfect for the day sailor who wants to take the next natural step and acquire the necessary knowledge and skills to be able to cruise, to sail about from place to place while living aboard.

All programs use boats that are brand new or less than two years old. The Jack Leverenz Sailing School does not sell boats, yacht ownership plans or tax advantages. Rather, all boats are owned and maintained by the Sailing School for the sole purpose of providing the best possible instruction in a location unequalled anywhere in the United States or Caribbean.

The professional instructors are full time employees and are excellent teachers as well as sailors. All are captains licensed by the U.S.C.G. and belong to appropriate professional Sailing Associations. Affiliated with the American Sailing Association, the Jack Leverenz Sailing School is authorized to grant official ASA ceritification to students of all four programs.

7-Day Basic Course

As a student, you'll board beautiful, roomy, new 34-foot boats for an adventure in learning. Your boat becomes your home for seven full days and nights as you learn to cruise under controlled conditions.

You'll sail in Charlotte Harbor, the finest protected sailing area in Florida, the Intracoastal Waterway and the Gulf of Mexico. A skilled instructor aboard helps you cruise among a variety of sandy beaches, islands, ports and coves as he teaches you the fundamentals of operating a cruising sailboat.

The sailboats used in the program are new 34 footers that have been modified slightly to Cruising School specifications to make them even more suitable for teaching cruising and for the areas in which they are used. The six and a half berths provide ample sleeping accommodations for four students and the instructor. The head has two doors, providing unusually good privacy for a boat of this size. The boats are completely equipped with all linens, bedding, galley and dining ware. All you need bring is your personal clothing.

Each boat has a crew of no more than four students and an instructor. Shipmates are arranged by the Cruising School, taking into consideration the goals, age, habits, and abilities of the students. A typical crew might be made up of two couples, four singles, one couple and two singles or a complete family. Occasionally a boat will go out with fewer than four students if there is not sufficient demand to fill each boat on a particular week. The common bond of a love of sailing and a thirst for adventure quickly turns strangers into true shipmates.

The only prerequisite for the 7-Day Basic Course is a general knowledge of basic sailing. This can be knowledge acquired by sailing one's own boat, having taken small boat sailing classes or even having read a good deal about sailing. In the case of couples, it is sufficient if one spouse has the necessary sailing experience. Starts every Saturday.

7-Day Advanced Course

You'll quickly go over the basics of your 34 or 39 footer as you get away early the first morning, to return again on the seventh afternoon. You'll stop at several of the more interesting islands and quaint harbors in the area, as more extensive navigation and an overnight passage are featured. Starts every Saturday.

10-Day Advanced Course

Includes everything in the 7-Day Advanced, plus a 140 mile non-stop passage to Key West and return. Other ports of call include Boca Grande, Cabbage Key, Captiva Island, Fort Myers Beach and Marco Island. Boats are new 39 footers. Starts every third Saturday and Wednesday.

14-Day Adventure in Paradise

No rigid itinerary, as the students participate in the planning of destinations up and down the southwest coast of Florida. Boats are new 39 footers and the course features extensive electronic and celestrial navigation. Starts on selected Saturdays.

It is expected that sailors new to cruising will take the 7-Day Basic Course first, before signing on for the Advanced Courses. If, however, one has some cruising background, such as overnighting on his own boat, or weekending on friends' boats, etc., it could be appropriate to start right out with an Advanced Course. Just as with the 7-Day Basic Course, in the case of couples, if one spouse has sufficient cruising experience, it's not necessary to have the same background.

Learn to Sail Programs

In addition to the Cruise Courses, comprehensive Learn-to-Sail Courses, featuring a fleet of 23-foot sailboats are also available.

MASTERCARD and VISA accepted

For complete information, send coupon or call TOLL FREE (800) 521-8802. In Michigan, (313) 886-7887.

JACK LEVERENZ SAILING SCHOOLS
17640 Mack Ave. • Grosse Pointe, MI 48224

Please send information about:

☐ Cruising Courses ☐ Sailing Courses

Name _____

Address _____

City / State / Zip _____

listing (Listing 1). If you LOAD and RUN the BASIC loader, it will clear the display, print a message showing where the loading is taking place, and when loaded will print out a message showing the entry points to the program. Once this is loaded it will remain in memory, out of the way of normal BASIC programs. To direct output to a parallel printer, type **SYS 49152**. A LIST command will now output *via* the parallel port as well as to the normal display screen. To stop output to the printer, type **SYS 49155**. It's as simple as that.

The parallel output may be going to a printer or to another computer. Depending on the nature of the other computer's parallel port, it may work with the basic printer parallel cable, or may require that the Strobe (STB-) and Acknowledge (ACK-) lines be swapped.

This program also allows the C64 to accept parallel input from another computer or device, and handles code conversions depending on whether or not the C64 is in the Upper or Lower case mode. To start accepting parallel input, type **SYS 49158**. Each character now received *via* the parallel port will be tested and, if necessary, converted. Parallel input will continue until either a C64 keyboard key is pressed or a **SYS 49161** is issued. With this parallel input setup, you can actually run your C64 remotely from another computer! It runs exactly as though your other computer was the C64 keyboard.

Conclusion

You can greatly expand the use of your C64 by using its built-in parallel port. It can save you money on your printer, eliminating the need for a serial port, and can allow you to bi-directionally connect your C64 to another intelligent device or computer. You can build on this initial parallel I/O program to interface to a world of parallel devices. Experiment!

Listing 2

```
20 REM PARALLEL I/O   ***  R.M.TRIPP  NOV 1983
21 READ MS:ME=MS:PRINT"{CLEAR}LOADING FROM";MS;"TO";
22 READ X:IF X<256 THEN POKE(ME),X:ME=ME+1:GOTO 22
23 PRINT ME-1
24 PRINT "{DOWN} OPEN PARALLEL OUTPUT  SYS 49152"
25 PRINT "CLOSE PARALLEL OUTPUT  SYS 49155"
26 PRINT "{DOWN} OPEN PARALLEL INPUT   SYS 49158"
27 PRINT "CLOSE PARALLEL INPUT   SYS 49161"
28 PRINT "                       OR PRESS ANY KEY"
10000 DATA 49152
10010 DATA 76,12,192,76,40,192,76,108
10020 DATA 192,76,136,192,173,38,3,141
10030 DATA 93,192,173,39,3,141,94,192
10040 DATA 169,53,141,38,3,169,192,141
10050 DATA 39,3,169,255,141,3,221,96
10060 DATA 173,93,192,141,38,3,173,94
10070 DATA 192,141,39,3,96,72,169,2
10080 DATA 44,24,208,240,27,104,48,13
10090 DATA 201,64,48,19,201,95,16,15
10100 DATA 24,105,32,208,10,201,192,48
10110 DATA 6,201,223,16,2,41,127,72
10120 DATA 104,32,95,192,76,0,0,72
10130 DATA 169,16,44,13,221,240,251,104
10140 DATA 141,1,221,96,173,36,3,141
10150 DATA 149,192,173,37,3,141,150,192
10160 DATA 169,151,141,36,3,169,192,141
10170 DATA 37,3,169,0,141,3,221,96
10180 DATA 173,149,192,141,36,3,173,150
10190 DATA 192,141,37,3,76,0,16,173
10200 DATA 198,0,208,236,169,16,44,13
10210 DATA 221,240,244,173,1,221,72,169
10220 DATA 2,44,24,208,208,7,104,201
10230 DATA 96,48,21,16,13,104,201,64
10240 DATA 48,14,201,95,16,4,9,32
10250 DATA 208,6,201,128,16,2,41,95
10260 DATA 108,38,3,256
```

[See page 63 for Listing Conventions]

Choosing the best investment in today's confusing financial world takes careful comparisons of the internal rates of return.

by Joseph Kattan

Requirements:
Any Atari Computer with minimum 24K RAM
Disk Drive

So you struck it rich in the bull market. That penny computer stock has surged from 1 and 1/16 to 4 and 1/4. And your boring utility stock has jumped from 10 to 23 while paying you a handsome 35 cent dividend every three months. Now which was the better gain? And what about that bond that your grandfather gave you on your seventeenth birthday to teach you the value of money?

Students of financial analysis learn early in their training to evaluate investments in terms of their "internal rate of return." I will get to the technical definition in a minute, but the internal rate of return is essentially a measure that allows you to compare different investments that have very little in common and learn the return they yield on your investment. It allows you to compare a $1,000 bond that yields an interest check every three months with a bond that pays interest only upon maturity or with a savings account into which you make a deposit or withdrawal whenever the spirit moves you or whatever else you might want to sink your money into. In other words, the measure will compare the rates of return on investments that generate uneven cash flows over uneven intervals.

The internal rate of return is defined in technical terms as the interest rate that equates the present value of the future receipts from an investment with the cost of the investment. To give a rough example, a note that costs $1,000 and will return $1,100 after one year has an internal rate of return of 10 percent. The formula for the internal rate of return is:

```
F(1)/(1+i)^1 +
 F(2)/(1+i)^2 + ...
  F(N)/(1+i)^N - I = 0
```

where I represents the initial investment, F represents an outlay or receipt, as appropriate, the exponents represent time periods, and i represents the internal rate of return. We can eliminate the factor I, however, by representing initial investment as a cash outlay, as follows:

```
I = - (F(0)/(1+i)^0)
```

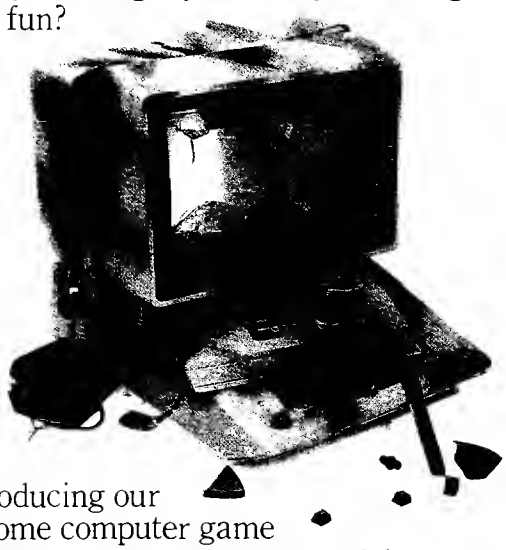
The problem in solving the internal rate of return should be evident from these equation. The equation cannot be solved for i. Rather, i has to be approximated, tested against the hypothesis that the left side of the equation equals zero, and then approximated again and again until a suitable approximation is obtained.

Calculating the internal rate of return by hand can be a time-consuming chore. With a long series of entries, the calculation requires numerous additions, divisions, and exponentiations, and if your initial approximation of the rate was off

Blow up

Gee whiz. Look what you've done. Your big shiny Apple, destroyed.

And you thought you were just having a little fun?



Introducing our new home computer game Cavern Creatures.™ Where one false move, one mistaken twitch of the Joystick can do terrible things to an Apple. Like shut parts down.

Who knows which ones? Making it act awful funny and then . . . KA-BOOM!

The game's finished. And maybe, so is your Apple. Oh, but don't worry.

Cavern Creatures is just a game. Full of action. Fun. Suspense. Just like our other games. Whether they're blowing up your Apple. Or totally blowing your mind.

How Much More Can One Joystick Jockey Take?

We've exceeded our past reputation for bringing you some of the most super snazzy, ultra-intense, graphically involving games around.

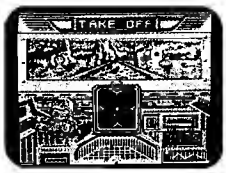
We've gone totally out of our minds with every game from shoot 'em up, blast 'em out of the sky strategies to mind game graphics guaranteed to provoke a mental meltdown.

Are you ready?

Cavern Creatures™

You can squirm. Beg. Plead. And moan. Nothing will save you, or your Apple, from being blown sky-high by this game. Don't worry. The effect's not permanent. Your Apple will recover. But will you?

your Apple.*



Space Ark™

Skin-hunters, poachers and mechanicals—enemy robots who aren't exactly the warm and friendly types—await you. Obliterate the bad guys. Save the good guys. You might survive this.

Ardy The Aardvark™

Here's a tongue that's as long and skilled as you are. Think you can lick stinging ants and tarantulas? Use Ardy's tongue as you make it through this maze!

Roundabout™

Sharpens your target skills with 24, count 'em, 24 different series of targets. How's your hand to eye coordination, pal?

Argos™

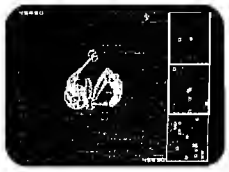
And if you like to push people around, lay off your little brother and take on these aliens instead. Save the Domed City from doomsville.

Super Bunny™

No, *Super Bunny* is not referring to all those Playboys stashed under the bed. It's a strategy/action game that just might turn your brains into carrot puree. Cover your burrow!

Conquering Worlds™

So you fancy yourself a die-hard strategist? Try conquering this. Maybe you'll rule the universe if mom will let you off restriction.



Bilestoad™

Avoid violence, but be prepared. Incredible graphics make this rated R! But if you're a wimp, pass this game up. It's not for babies.

DATAMOST

The most out of our minds.™

target, the same additions, divisions, and exponentiations again. In other words, a perfect problem for the computer to solve. The computer does not mind looping over and over again until it approximates the correct answer. And with The Investor, you simply plug in your numbers and wait for the Atari to do the real work.

Before getting into the technical details of the program, here are some instructions on running it. The Investor is a menu-driven program with simple commands. The first program menu gives you three choices: examining the disk directory for data files created by the program, creating data files, and running calculations in an "immediate" mode.

Disk Data Files

To create a disk data file, choose menu option 2. You will then be asked to enter the name of the file, the amount of the initial investment, and the date of that investment. There are some restrictions on file names. The file name may contain up to eight characters -- upper case letters and numbers are permitted -- and must begin with an upper case letter. If you enter an impermissible file name, The Investor will so advise you and ask you to reenter the file name. Enter the date of the investment in a month, day, year order. The date entry routine will automatically skip over the slashes separating month from the day, and the day from the year. You may use the backspace key, but the remaining cursor control keys have been disabled.

Once you have entered that information, you will be given the option of adding transactions to your data file or writing the data to disk. You may interrupt this part of the program, or any part of the program, by pressing the OPTION key at any point when the program allows you to enter information. If, for example, you entered an incorrect file name, you may press the OPTION key when asked for the date. This will return you to the main menu, but at a cost; all of the data entered during that part of the program will be erased from memory.

Once you have created the data files, choose the directory menu option to revise or update your files. That menu option will display a listing of all the data files on your disk and allow you to examine any file by entering its number from a menu. You may then review the current internal rate of return on your investment or update the data file on that account.

A bit of advice on updating data files. You may update a data file with only two types of transactions: an investment -- think of it as the equivalent of a deposit into a saving account, or a receipt -- think of it as the equivalent of a withdrawal from a savings account. Do not write to disk any information other than the equivalent of a deposit or a withdrawal. For example, if your bank credited your account with $30 in interest, that information should not be written to the disk -- the money is still in your account and will show up in the current account balance. If, on the other hand, the bank sent you a check for $30, write that to the disk; the money is no longer in your account.

Once you have updated your files, you may review the internal rate of return. When you choose the review option, you will be asked for the current value of the account and the current date. The screen will then go blank, while the program calculates the

internal rate of return. The screen will come alive again when the calculation has been completed or if the program encounters invalid data. In both cases, the computer's buzzer will advise you that the computations have been completed.

Because of the numerosity of the calculations, expect the program to take at least 20 seconds to calculate the rate of return. Of course, the greater the number of transactions to evaluate, the more time the program takes. Remember that the program must perform approximately two additions, one division, and one exponentiation for each transaction per loop and that it will take anywhere from 10 to 20 loops to come up with the correct answer.

You must use the disk file option if your investments and receipts are of irregular amounts and occur in irregular intervals. The Investor will display all of your transactions and their dates, and a running total of your investments and receipts for each account. If you wish to determine the rate of return on a one-time investment, however, you may use the immediate mode, option 3 on the main menu.

The Immediate Mode

Menu option 3, "Rate of Return," allows you to examine the internal rate of return on a one-time investment. For example, if you are purchasing a bond that will pay you a specific amount every three months and the face value at its expiration, you may use this mode. The Investor will ask you for five items of information: your initial investment (the cost of the bond in the example above), the amount per receipt (the regular interest check you will receive), the interval of the receipt (from an option of monthly, quarterly, or annual receipts), and the return at the end of the investment (the face value of the bond in the example above). Once you have entered this information, the screen will go blank and the program will calculate your internal rate of return.

Let us take as an example a six-month note that you purchased for $10,000. At the end of every month, you will receive a check for $100 and at the end of the term you will receive your initial $10,000 back. You would enter 10000 for the investment, 100 as the amount per receipt, 6 for the number of receipts, monthly as the interval of the receipts, and 10000 as the return at the end of the term. The program screen will look like this:

THE INVESTOR

INVESTMENT: *10000*
AMOUNT PER RECEIPT: *100*
NUMBER OF RECEIPTS: *6*

INTERVAL OF RECEIPTS:
1) MONTHLY 2)QUARTERLY
3) ANNUAL *1*

RETURN AT END OF TERM: *10000*

INTERNAL RATE OF RETURN: 12.69%

The number in italics represents the information that you enter into the program. The Atari takes off from there and calculates your internal rate of return on the investrnent.

Explanation of Program

The Investor contains a number of subroutines that may be of interest to you even if you have little or no interest in computing the internal rate of return on investments. In particular, the data input routine, the date input routine, the disk directory routine, and the decimal justification routine may come in handy in other programs.

You will notice that the INPUT statement is never used in The Investor. Instead, the subroutine at lines 300 through 312 accepts individual key strokes, tests them, and accepts only valid ones. For that reason, the cursor control keys will have no effect on the screen display. If the program expects numerical information, it will ignore all keys except the numbers 0 through 9 and the period mark. The subroutine at line 240 is used for date entry. It prints a mask on the screen and then GETs individual keystrokes, tests them, and prints them on the screen only if they represent the numbers 0 through 9. The seemingly superfluous POKE at line 310 and the numerical array in the data entry subroutine are designed to correct a quirk in Atari BASIC. For some reason, the Atari BASIC interpreter will change the contents of strings when a buffer is open to the Atari's keyboard. (For another solution to this problem, see "Banish Atari INPUT Statements," in the August 1983 Compute!).

The disk directory routine at line 1000 could easily be changed to a program menu by removing the restrictions on the listings read from the directory. Finally, the decimal justification subroutine beginning at line 425 deserves mention. The subroutine will round numbers to the nearest cent, convert them into strings, insert the trailing zeros on integers or numbers ending with multiples of 0.1, and then work itself backward to insert commas every three spaces, beginning from the location of the decimal. When the routine runs out of numbers, a TRAP statement forces it into a line containing the PRINT instruction.

The program's main loop begins at line 3300. It is here that some modifications of the program might prove useful to suit your individual needs. The loop works by beginning with three guesses of the rate of return: R --the expected return, R1 -- the lower limit, and R2 -- the upper limit. After each loop, the program exchanges these values. If R -- the actual guess --was too low, then R1 -- the lower limit -- is raised to the value of R, and R is raised to a value between its current value and R2 -- the upper limit. R1 and R2 are initialized at very safe values -- to allow the program to compute extreme rates of return, such as those obtained when you doubled your money in two months. Because of this "safe" initialization, however, the program must loop around more times than need be in the more typical case of returns between 5 and 15 percent. One solution is to delete line 3335. The other is to ask the user to input R, R1, and R2. Listing 2 contains that modification. Beware, however, that a bad entry could send the program into an endless loop. In addition, Listing 2 does not control the integrity of the screen display as tightly as does the rest of the program. You could also change the values in line 3345 for jumping out of the loop. Those values require that two successive approximations of the internal rate of return be within 5 one-hundredths of one percent of each other, and that the left side of the internal rate of return equation (which optimally will be zero) is less than 0.0005. You could raise these values a little bit, particularly since the Atari's faulty exponentiation precludes perfect results (ask the Atari to PRINT 2 ↑ 3 to see what I mean).

Finally, you should note that The Investor will perform the rate of return computations on up to 360 transactions. Be warned, however, that you will wait quite a long time to see the results if your record is that long.

Listing 1

```
5 POKE 16,64:POKE 53774,64:GOTO 5000:REM DISABLE BREAK KEY
50 A$="NUM":GOSUB 300:IF N1<1 OR N1>MX THEN POSITION C,R:GOTO 50
55 RETURN
80 PRINT "WRITE TO DISK ";:GOSUB 175:IF A$="Y" THEN GOSUB 150
90 PRINT "{UP}";:RETURN
100 RESTORE :FOR N1=1 TO M:READ DAYS:NEXT N1:
    REM THIS SUBROUTINE CALCULATES THE NUMBER OF DAYS BETWEEN TWO DATES
110 DAYS=DAYS+Y*365+INT(Y/4)+D+1-INT(Y/100)+INT(Y/400)
120 IF Y/400=INT(Y/400) AND M>2 THEN 5735
130 IF Y/100=INT(Y/100) THEN DAYS=DAYS-1
140 IF J=1 THEN DAY0=DAYS
145 RETURN
150 TRAP 155:OPEN #1,8,0,ACT$:FOR J=1 TO I:PRINT #1;F(J):
    PRINT #1;D(J):PRINT #1;DATE$(J*8-7 J*8):NEXT J
152 CLOSE #1:RETURN
155 CLOSE #1:TRAP 5010:IF PEEK(195)<>162 THEN PRINT
    "{BEEP}PROBLEMS WITH DISK DRIVE. TRY AGAIN.":GOTO 165
160 PRINT "{BEEP}DISK FULL. PLEASE INSERT NEW DISK."
165 GOSUB 565:GOTO 150
175 PRINT " (yES OR nO)? ";
180 GOSUB 300:A$=Q$(1,1):IF A$="Y" OR A$="N" THEN
    PRINT "{UP,DELETE LINE}";:RETURN 185 POSITION C,R:GOTO 180
200 PRINT "{CLEAR}":POKE 85,14:PRINT "THE INVESTOR":PRINT :RETURN
240 N=PEEK(85)-1:REM DATE ENTRY ROUTINE-ENTER DAY,MONTH,YEAR
241 TRAP 241:POSITION N+1,S:PRINT "{CTRL E}-/--/----":J=1:A$=""
242 POSITION N+J+(J>2)+(J>4),S:PRINT "{CTRL E}":IF PEEK(53279)=3 THEN 312:
    REM ON OPTION KEY CLEAR MEMORY AND GO TO MENU
243 IF PEEK(764)=255 THEN 242
244 GET #5,A:IF A=126 AND J<>1 THEN POSITION N+J+(J>2)+(J>4),S:
    PRINT "-":J=J-1:GOTO 242
245 IF A<48 OR A>57 THEN 242
246 POSITION N+J+(J>2)+(J>4),S:PRINT CHR$(A):S(J)=A:
    J=J+1:IF J<9 THEN 242
248 GET #5,A:IF A=126 THEN J=8:GOTO 242
250 IF A<>155 THEN 248
252 FOR J=1 TO 8:A$(J,J)=CHR$(S(J)):NEXT J:M=VAL(A$(1,2)):
    D=VAL(A$(3,4)):Y=VAL(A$(5,8))
254 IF M=0 OR M>12 OR D=0 OR D>31 OR (M=2 AND D>29) OR
    (M=2 AND Y/4<>INT(Y/4) AND D>28) THEN 256
255 TRAP 980:POSITION 2,22:PRINT "{DELETE LINE}":POSITION 2,S+1:RETURN
256 POSITION 2,22:PRINT "{BEEP,DELETE LINE}PLEASE RE-ENTER CORRECT DATE":GOTO
241 260 IF M<10 THEN DATE$(I*8-7,I*8-7)="0":
    DATE$(I*8-6,I*8-6)=STR$(M):GOTO 270
265 DATE$(I*8-7,I*8-6)=STR$(M)
270 IF D<10 THEN DATE$(I*8-5,I*8-5)="0":
    DATE$(I*8-4,I*8-4)=STR$(D):GOTO 280
275 DATE$(I*8-5,I*8-4)=STR$(D)
280 DATE$(I*8-3,I*8)=STR$(Y):RETURN
300 C=PEEK(85):R=PEEK(84):
    REM THIS SUBROUTINE IS USED IN LIEU OF INPUT STATEMENTS
301 TRAP 301:Q$="":POSITION C,R:PRINT BL$;BL$:Y=1:S=32
302 POKE 764,255:A=0:POSITION C+Y-1,R:PRINT "{CTRL E}";
303 IF PEEK(53279)=3 THEN 312:
    REM ON OPTION KEY,CLEAR MEMORY AND GO TO MENU
304 IF PEEK(764)=255 THEN 303
305 GET #5,A:IF A=155 AND Y<>1 THEN 310
306 IF A=126 THEN Y=Y-(Y>1):Q$=Q$(1,Y):POSITION C+Y,R:
    PRINT " ":GOTO 302
307 IF A<32 OR A>124 OR (A$="NUM" AND (A<48 OR A>57) AND A<>46) THEN 302
308 Q$(Y)=CHR$(A):POSITION C+Y-1,R:PRINT CHR$(A):IF Y=1 THEN S=A
309 Y=Y+1:GOTO 302
310 POKE ADR(Q$),S:PRINT CHR$(126):IF A$="NUM" THEN N1=VAL(Q$):A$="    "
```

```
311 RETURN
312 PRINT "{BEEP}":CLR :GOTO 5005
314 PRINT "NAME OF FILE: ";
315 GOSUB 300:S=LEN(Q$):IF S>0 AND S<9 THEN 322
320 POSITION 2,21:PRINT "{UP,DELETE LINE,BEEP}FILE NAME MUST BE 1-8 CHARACTERS
    LONG":POSITION C,R:GOTO 15
322 IF ASC(Q$(1,1))<65 OR ASC(Q$(1,1))>90 THEN 336
325 FOR Y=2 TO LEN(Q$):S=ASC(Q$(Y,Y)):
    IF (S<65 OR S>90) AND (S<48 OR S>57) THEN POP :GOTO 335
330 NEXT Y:ACT$="D:":ACT$(3)=Q$:ACT$(LEN(ACT$)+1)=".IRR":RETURN
335 POSITION 2,21:PRINT "{UP,DELETE LINE,BEEP}NAME MAY CONTAIN UPPER CASE
    LETTERS OR NUMBERS ONLY":POSITION C,R:GOTO 315
336 POSITION 2,21:PRINT "{UP,DELETE LINE,BEEP}NAME MUST BEGIN WITH UPPER CASE
    LETTER":POSITION C,R:GOTO 315
340 PRINT "INVESTMENT: $";:GOSUB 346:F(I)=INT(-N1*100+0.5)/100:
    MI=MI+F(I):RETURN 345 PRINT "RECEIPT: $";:GOSUB 346:
    F(I)=INT(N1*100 0.5)/100:PL=PL+F(I):RETURN
345 PRINT "RECEIPT: $";:GOSUB 346:F(I)=INT(-N1*100+0.5)/100:MI=MI+F(I):RETURN
346 A$="NUM":GOSUB 300:IF N1<10000000 THEN POKE 84,22:PRINT "{UP,DELETE LINE}":
    POKE 84,R+1:RETURN
347 POKE 84,22:PRINT "{UP,DELETE LINE,BEEP}AMOUNTS CANNOT EXCEED 9,999,999":
    POSITION C,R:GOTO 346 350 PRINT "DATE OF TRANSACTION: ";
355 S=PEEK(84):GOSUB 240:GOSUB 260:RETURN
425 N=INT(N*100+0.5)/100:Z$=BL$(1,13):N$=STR$(ABS(N)):
    IF N=INT(N) THEN N$(LEN(N )+1)=".00":GOTO 435
427 REM SUBROUTINE AT 425 JUSTIFIES DECIMAL AND INSERTS COMMAS
    IN NUMBERS OVER 99.99
428 REM BEFORE JUMPING TO SUBROUTINE, CHANGE VARIABLE CONTAINING
    NUMBER TO BE JUSTIFIED TO N
430 IF N*10=INT(N*10) AND N<100000000 AND LEN(N$)<=10 THEN N$(LEN(N$)+1)="0"
435 Z$(11,13)=N$(LEN(N$)-2):N$=N$(1,LEN(N$)-3):L=(LEN(N$)-1)/3:DIST=3:PT=3
445 IF L<1 THEN 455
450 Z$(11-DIST,13-DIST)=N$(LEN(N$)-PT+1,LEN(N$)-PT+3):
    Z$(10-DIST,10-DIST)=",":L L-1:DIST=DIST+4:PT=PT+3:GOTO 445
455 DIST=DIST-4:PT=PT-3
460 TRAP 465:Z$(9-DIST,9-DIST)=N$(LEN(N$)-PT,LEN(N$)-PT):
    DIST=DIST+1:PT=PT+1:GOTO 460
465 POKE 85,20:PRINT Z$:RETURN
565 PRINT :POKE 85,7:PRINT "PRESS return TO CONTINUE"
570 POKE 764,255
571 IF PEEK(53279)=3 THEN 312
572 IF PEEK(764)=255 THEN 571
574 GET #5,A:IF A=155 OR (Y=100 AND A=42) THEN Y=0:RETURN
575 GOTO 570
970 PRINT :PRINT :POKE 85,10:PRINT "PRESS RETURN FOR MENU":
    GOSUB 570:GOTO 5010
980 POKE 559,34:PRINT :PRINT "{BEEP} INVALID DATA ENTERED":GOTO 970
990 PRINT "{CLEAR}":PRINT "THE INVESTOR: ";ACT$(3,LEN(ACT$)-4):PRINT:RETURN
1000 TRAP 1070:GOSUB 200:OPEN #1,6,0,"D:*.*":PRINT :S=35:GOSUB
1095:N=0:TRAP 1040:R M SUBROUTINE READS THE DISK
1005 REM DIRECTORY AND LISTS ALL FILES WITH THE IRR EXTENSION
1010 INPUT #1;Q$:IF Q$(11,13)<>"IRR" THEN 1010
1015 DATE$((N+1)*12-11,(N+1)*12)=Q$(3,14)
1020 N=N+1:Y=(N/2=INT(N/2)):POKE 85,2+20*Y:PRINT N;")";:POKE 85,6+20*Y:
     PRINT Q$(3,10);: F Y=0 THEN POKE 85,19:PRINT "!";:GOTO 1010
1030 PRINT :GOTO 1010
1040 CLOSE #1:IF Y=0 AND N<36 THEN PRINT
1050 GOSUB 1095:IF N=0 THEN 1100
1060 ACT$="D:":PRINT "ENTER NUMBER OF FILE TO LOAD: ";
1070 MX=N:GOSUB 50
1080 Q$=DATE$(N1*12-11,N1*12):DATE$=" ":DATE$(960)=DATE$:DATE$(2)=DATE$
1085 FOR I=1 TO 8:IF Q$(I,I)<>" " THEN ACT$(LEN(ACT$)+1)=Q$(I,I)
```

```
1090 NEXT I:ACT$(LEN(ACT$)+1)=".IRR":GOSUB 1200:GOSUB 990
1091 FOR C=1 TO I-1:PL=PL+F(C)*(F(C)>0):MI=MI+F(C)*(F(C)<0):NEXT C
1092 PRINT "UPDATE OR REVIEW? ";:GOSUB 300:A$=Q$(1,1):IF A$<>"U" AND A$<>"R"
THEN PRINT "{UP}";:GOTO 1092
1093 PRINT "{UP,DELETE LINE}";:IF A$="U" THEN 2000
1094 GOTO 3110
1095 FOR C=0 TO S:PRINT "{CTRL R ";:NEXT C:PRINT :RETURN
1100 PRINT :PRINT "{BEEP}THIS DISK HAS NO DATA FILES FOR THIS PROGRAM.
     PLEASE INSERT CORRECT DISK AND PRESS return.";
1110 PRINT "  PRESS * TO CREATE DATA FILES, option FOR MENU":
     Y=100:GOSUB 570 1120 IF A=42 THEN 4000
1130 GOTO 1000
1200 TRAP 1220:OPEN #1,4,0,ACT$:I=1
1210 INPUT #1;N:F(I)=N:INPUT #1;N:D(I)=N:INPUT #1;Q$:
     DATE$(I*8-7,I*8)=Q$:I=I+1: OTO 1210
1215 REM READ RECORDS UNTIL AN END OF FILE ERROR
1220 CLOSE #1:REM CLOSE IOCB ON ERROR--FILE NOW COMPLETE
1230 IF PEEK(195)<>136 THEN 1270:REM IF NOT END OF FILE ERROR,
     IT MUST BE SOME THEIR ERROR
1235 IF I<12 THEN N1=I-1:GOTO 1260
1240 PRINT "{UP,DELETE LINE}";I-1;" RECORDS AVAILABLE. UP TO 11 MAY BE VIEWED
     AT ONE TIME. ENTER NUMBER OF RECORDS TO DISPLAY ";
1250 MX=I-1:PRINT "(1 TO ";MX;"): ";:GOSUB 50
1260 T=I-N1:RETURN
1270 PRINT :PRINT "{BEEP}PROBLEMS WITH DISK DRIVE. ENTER AGAIN":GOTO 970
2000 GOSUB 3010:PRINT "iNVESTMENT OR rECEIPT? ";
2010 GOSUB 300:IF Q$(1,1)<>"I" AND Q$(1,1)<>"R" THEN POSITION C,R:GOTO 2010
2015 PRINT "{UP,DELETE LINE}";:IF Q$(1,1)="I" THEN GOSUB 340:GOTO 2030
2020 GOSUB 345
2030 GOSUB 350
2035 POKE 84,R+1:PRINT "{UP,DELETE LINE2}MORE REVISIONS";:GOSUB 175:IF A$="Y"
THEN I=I+1

2040 IF I>11 THEN T=I-10
2045 GOSUB 990:IF A$="Y" THEN 2000
2050 A$="GO":GOSUB 3010:GOSUB 80:PRINT :PRINT "REVIEW OR MENU? ";
2065 GOSUB 300:A$=Q$(1,1):IF A$<>"R" AND A$<>"M" THEN POSITION C,R:GOTO 2065
2070 IF A$="R" THEN I=I+1:GOSUB 200:PRINT ACT$(3,LEN(ACT$)-4):PRINT :GOTO 3110
2080 GOTO 5100
2500 GOSUB 200:PRINT :PRINT :PRINT "INVESTMENT: ";:A$="NUM":
     GOSUB 300:F(1)=-N1:D(1)=0:MI=F( )
2510 PRINT "AMOUNT PER RECEIPT: ";:A$="NUM":GOSUB 300:F(2)=N1:
     PRINT "NUMBER OF RECEIPTS ";:A$="NUM":GOSUB 300:G=N1
2520 PRINT :PRINT "INTERVAL OF RECEIPTS:";
     PRINT "1) MONTHLY 2) QUARTERLY 3) ANNUAL ";:MX= :GOSUB 50:PRINT
2530 N=1/12*(N1=1)++0.25*(N1=2)+(N1=3)
2540 PRINT "RETURN AT END OF TERM: ";:A$="NUM":GOSUB 300:F(G+2)=N1:PL=F(G+2)
2550 FOR I=2 TO G+1:F(I)=F(2):PL=PL+F(2):D(I)=D(I-1)+N:NEXT I:D(G+2)=D(G+1)
2560 TRAP 980:G=D(G+2):S=1.0E+97:GOTO 3300
3010 FOR J=T TO I-1+A$="GO":PRINT DATE$(J*8-7,J*8-6);"/";
     DATE$(J*8-5,J*8-4);"/";DAT $(J*8-3,J*8);
3020 POKE 85,14:IF F(J)<0 THEN PRINT "INV.";:GOTO 3040
3030 PRINT "REC.";
3040 N=F(J):GOSUB 425
3050 IF PEEK(84)>13 AND J<I-1 THEN GOSUB 565:GOSUB 990
3060 NEXT J:S=30:GOSUB 1095:PRINT "  TOTAL INVESTED";:N=ABS(MI):
     GOSUB 425:PRINT " TOTAL RECEIVED";:N=PL:GOSUB 425
3070 S=30:GOSUB 1095:R=PEEK(84):A$="":RETURN
3110 GOSUB 3010
3115 PRINT "ENTER CURRENT VALUE $";:GOSUB 346:F(I)=N1:PL=PL+F(I)
3120 PRINT "ENTER TODAY'S DATE: ";:GOSUB 355
3300 PRINT :POKE 85,10:PRINT "ONE MOMENT PLEASE":POKE 20,0
```

```
3305 IF PEEK(20)=90 THEN POKE 559,0:GOTO 3315+10*(S=1.0E+97):
     REM POKE TURNS OFF SCREEN DISPLAY TO SPEED UP COMPUTATION
3310 GOTO 3305
3315 FOR J=1 TO I:M=VAL(DATE$(J*8-7,J*8-6)):D=VAL(DATE$(J*8-5,J*8-4)):
     Y=VAL(DATE$(J*8-3,J*8)):GOSUB 100
3320 D(J)=(DAYS-DAY0)/365:NEXT J:G=D(I)
3325 R1=-1*(PL<ABS(MI)):R=(ABS(PL/MI)-1)/G:R2=ABS(PL/MI)*(PL>ABS(MI)):
N1=0:N2=0
3330 REM FOR FASTER EXECUTION. BUT WITH GREATER LIMITATIONS, DELETE LINE 3335
3335 IF G<1 AND PL>ABS(MI) THEN R2=(R+1)^(1/D(2))
3340 N=0:FOR J=1 TO I:N=N+F(J)/((1+R)^(D(J))):NEXT J:N2=N1:N1=N:
     IF N2=0 THEN N2 N1
3345 IF ABS(F(0)-R)<5E-04 AND N<5E-04 THEN PRINT "{UP,DELETE LINE}";:
     POKE 559,34:GOTO 3390
3350 G=1:IF SGN(N2)=SGN(N1) AND N2<>N1 THEN G=ABS(N1/(N2-N1)):GOTO 3360
3355 IF N2<>N1 THEN G=(ABS(N1)-ABS(N2))/ABS(N1)
3360 S=G+1:IF N<0 THEN R2=R:R=(R+G*R1)/S:F(0)=R2
3365 IF N>0 THEN R1=R:R=(R+G*R2)/S:F(0)=R1
3370 GOTO 3340
3390 PRINT :POKE 559,34:PRINT "{BEEP}INTERNAL RATE OF RETURN: "
     ;INT(R*10000+0.5)/100;"%":GOTO 970
4000 GOSUB 200:GOSUB 314:I=1:GOSUB 340:GOSUB 350:PRINT
4005 PRINT "return FOR MENU, * FOR MORE ENTRIES":Y=100:GOSUB 570
4010 IF A=42 THEN PRINT "{UP,DELETE LINE,UP,DELETE LINE}":GOSUB 200:I=2:T=1:
     PRINT ACT$(3,LEN(ACT$)-4):PRINT :GOTO 4015 PRINT :GOSUB 80:
     PRINT :GOTO 5100
5000 POKE 710,160:POKE 712,148:POKE 752,1:OPEN #5,4,0,"K:"
5005 DIM ACT$(15),F(120),DATE$(960),D(120),BL$(24),N$(13),Z$(14),
     A$(9),Q$(15),S 8)
5010 ACT$="":DATE$=" ":DATE$(960)=DATE$:DATE$(2)=DATE$:N$="":Z$="":A$="":Q$=""
5020 BL$=" ":BL$(23)=BL$:BL$(2)=BL$
5025 FOR I=1 TO 120:F(I)=0:D(I)=0:NEXT I
5100 PL=0:MI=0:GOSUB 200:PRINT :PRINT "(1) DISPLAY DISK DIRECTORY":
     PRINT "(2) CREATE DISK FILE":PRINT " 3) RATE OF RETURN"
5110 MX=3:GOSUB 50
5120 ON N1 GOTO 1000,4000,2500
5790 DATA 0,31,59,90,120,151,181,212,243,273,304,334
```

MICRO™

Listing 2

```
3300 PRINT :PRINT "APPROXIMATE INTEREST RATE (%): ";:
     A$="NUM":GOSUB 300:T=N1/100
3305 PRINT "UPPER LIMIT: ";:A$="NUM":GOSUB 300:R2=N1/100:
     IF R2<T THEN PRINT "UPPER LIMIT MUST BE HIGHER":GOTO 3305
3307 PRINT "LOWER LIMIT (%): ";:A$="NUM":GOSUB 300:R1=N1/100:
     IF R1>T THEN PRINT "LOWER LIMIT MUST BE LOWER":GOTO 3307
3310 PRINT "{UP,DELETE LINE,UP,DELETE LINE,UP,DELETE LINE}":
     POKE 85,10: PRINT "ONE MOMENT PLEASE":POKE 20,0
3315 IF PEEK(20)=90 THEN POKE 559,0:GOTO 3325+10*(S=1.0E+97):
     REM TURN OFF SCREEN DISPLAY TO SPEED UP COMPUTATION
3320 GOTO 3315
3325 FOR J=1 TO I:M=VAL(DATE$(J*8-7,J*8-6)):
     D=VAL(DATE$(J*8-5,J*8-4)):Y=VAL(DAT $(J*8-3,J*8)):GOSUB 100
3330 D(J)=(DAYS-DAY0)/365:NEXT J:G=D(I)
3335 R=T:N1=0:N2=0
```

MICRO™

MICRO™

From Here to Atari

by Paul S. Swanson

SofToy

About a year or more ago, I purchased a copy of My First Alphabet, which nicely supplemented material my young daughter was learning from Sesame Street. The alphabet and the numbers through nine are no mystery to her. She is now 2-1/2 and still enjoys the program, but in looking for other programs for her to try, I came across SofToy, a product of Monarch Data Systems, Inc. (P.O.Box 207, Cochituate, MA 01778). The object of this game is, like My First Alphabet, teaching the letters. In SofToy, however, a much different approach is used.

The first of several levels in the program teaches keyboard zones. Nine pictures on the screen can be animated by pushing a key in the corresponding zone on the keyboard. An appropriate sound accompanies the animation. For example, the sound of the train whistle accompanies smoke pouring out of the smokestack. The next level uses the same nine pictures with numbers, performing the animation routine in response to the corresponding number key.

The levels advance next to random letters below each picture. At various times, the screen lights up and makes an easily distinguishable sound and a new set of letters appears below the picture. As with the numbers, the pictures are animated in response to pressing the corresponding keyboard key. There are also two variations of the numbers and letters schemes. One is program mode, where you can enter a string of letters or numbers, then hit RETURN and the computer will play the indicated pictures in that order. The second variation is a match game. Once the match game is set up, two pictures, one at a time, will animate. The object is to press the keys, in the same order, which correspond to those two pictures.

In contrast to My First Alphabet, SofToy is a much easier program for a preschooler to master. There are no wrong buttons on the computer at all. On My First Alphabet, SYSTEM RESET will cause a reboot and the function keys are used to display and control a menu of selections, requiring reading on the part of the operator. SofToy has no menu. The only words on it really don't have to be read in order to operate the game and the screens indicate which mode the game is in much more obviously than does My First Alphabet.

Printing

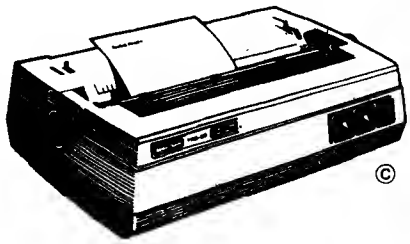
I recently acquired an Atari 1027 letter quality printer. Although I reviewed it when I announced the new Atari product line, I have now had some practical hands on experience with it. The printer is produced by Atari and the list price is $349.95. In contrast to dot matrix printers, this 1027 is very small - barely enough room to accommodate the paper. It is about the same width as my Epson MX-80, but only half as deep and about 3/4 as high. I found this to be a convenient feature. The printer can be placed almost anywhere. No longer does a large supply of continuous form need to be placed under the printer, with enough room for long listings to scroll off onto the floor behind it. The 1027 is designed for single sheets, so no room for paper is really required at all. I use it with continuous form, ripping the sprocket holes off so that the paper will fit. Single sheets tend to slip when the end of the sheet gets past the initial rollers, which isn't a problem with the continuous form. The printer is very slow, which is a drawback in many ways. I will admit that part of the satisfaction of this new acquisition is that I also have the faster Epson printer for listings and drafts. However, for occasional use, this very inexpensive letter quality printer is a printer worth considering.

The Atari 1027 printer produces good quality output when used within its limitations. In addition to the problems with slipping when using single sheets, I have also had problems with mailing labels. If the paper is too heavy, the surface is too soft for the printer and I have yet to discover any adjustment to compensate, so mailing labels didn't work. Many of the letters simply didn't legibly print. The printer is excellent for letters, reports and anything else that requires letter width paper, which may be ideal in places like a college or university for use in word processing, producing the multitude of written work required.

New Topics

Many of the topics covered in this column are based on letters received. If you would like to see any particular topic covered in this column, send a letter to me or, if you have a modem on your computer, you can leave it as a message on Nite Lite or in the Atari World section of the Outpost, which is at (617) 259-0181.

MICRO™



Raise your Apple's IQ

Six Times A Year!

A One Year Subscription Brings You 12 Issues With:

Over $500 of Programs for your Home, Business, Education and Entertainment. Complete Program Listings with Instructions.

Comprehensive Articles that show what each program does, how to use it and how to type it into your Apple, Franklin ACE or other Applesoft-compatible computer.

Regular Features for the Beginner and Expert.

On The Scene
The Latest New Software/Hardware Releases.

Products! Inside and Out
Comprehensive Product Reviews.

Education Corner
Programs that help make Learning Fun.

Tips 'N Techniques
Little known programming Tricks you can Use.

Disassembly Lines
An Expert reveals the mysteries of Applesoft.

Utilities
Superchargers for Basic, DOS, Printing, and More.

Games
Arcade Fun you can Type and Run.

Note
☐ Domestic U.S. First Class subscription rate is $51.95
☐ Canada Air Mail subscripton rate is $59.95
☐ Outside the U.S. and Canada Air Mail subscription rate is $89.95
All payments must be in U.S. funds drawn on a U.S. bank.

Try a NIBBLE!

Here's what some of our Readers say:

☐ *"Certainly the best magazine on the Apple!"*
☐ *"Impressed with the quality and content."*
☐ *"Programs remarkably easy to enter."*
☐ *"I'll be a subscriber for life!"*
☐ *"Your service is fantastic . . . as a matter of fact, I'm amazed!"*

Try a NIBBLE!

NIBBLE is focused completely on the Apple and Applesoft-compatible computers.

Buy NIBBLE through your local Apple Dealer or subscribe now with the Coupon or Order Card in this issue.

You'll want Back Issues Too!

Here are some examples of programs you can get:

The Investor—Stock Tracking, Reporting, and Graphing.

Recipe Box—Kitchen/Menu Management made Fun.

The Librarian—Auto Logging and Retrieval of your Disks.

Designer/Illustrator—Art/Design Creation and Composition with Graphics.

Machine Language Editor—Quick and Easy Aid for Typing and Changing M/L Programs.

And Much . . . Much More!
NIBBLE will become a permanent part of your Reference Library. Discover why 95% of NIBBLE Readers save every issue!

Join more than 120,000 Apple/Ace users who say:
"NIBBLE IS TERRIFIC!"

SUBSCRIBE NOW AND SAVE $12.00 OFF THE COVER PRICE!

nibble

MasterCard VISA
We accept Master Charge & Visa

Box 325, Lincoln, MA 01773 (617) 259-9710

I'll try nibble!
Enclosed is my $26.95 (for 12 issues)
(Outside U.S., see special note on this page.)

☐ **check** ☐ **money order** ☐ **bill me (U.S. only)**

Your subscription will begin with the next issue published after receipt of your check/money order.

Card # _____ Expires _____
PLEASE PRINT CLEARLY

Signature _____

Name _____

Address _____

City _____

State _____ Zip _____

Fast Low Cost A/D Converter

Frederick J. Genett

Requirements:
'Build-it-yourself' A/D converter
Any microcomputer with a parallel port

Perhaps you are an old pro at bridging the gap between the analog world we live in and the digital domain inhabited by your microcomputer. Or perhaps you are like the rest of us, a little hesitant about plugging things into the back of your computer. The selection of A/D converters is confusing at best. One must consider such things as resolution, speed, accuracy, cost and ease of interfacing.

Recently a chip has been introduced onto the market which should lessen the nightmare of A/D selection for most applications. This chip, National Semiconductor's **ADC0820**, boasts 1.2 microsecond maximum conversion time, ± 1 LSB total unadjusted error, 35 milliwatts typical power dissipation, and an 8-bit parallel, bus compatible output. A couple of years ago an A/D converter exhibiting similar characteristics would cost about two hundred ($200) collars. The ADC0820 presently sells for about fifteen dollars in single unit quantities.

Construction

The beauty of the ADC0820 is exemplified in the ease with which it is interfaced. The circuit in Figure 1 has been used with both the PET and VIC-20 computers and can in fact be connected to any computer having a latched data bus. Power can be supplied externally, from pin 2 of the cassette port or, in the case of the VIC-20, from pin 2 of the User Port.

In the circuit of Figure 1, Vref+ is tied to 5 Vdc and Vref- to ground. This will provide counts of 0 and 255 for Vin equal to 0 and 5 volts respectively. If, however, your application requires a full scale voltage less than 5 Vdc or a count of 0 to be given for a Vin offset from ground, Vref+ and Vref- can be adjusted accordingly.

Circuit construction can be printed circuit, wire wrap or point-to-point wiring. If wire wrapping is used, hookup wire should be soldered from the analog input signal to Vin. Whichever method is used, this connection should be kept as short as possible. Finally, since, the ADC0820 is a CMOS device, care should be taken to avoid static electricity.

Software

Listings of the machine language loader routines are shown below for a variety of microcomputers. These represent two programs. Each program will load the machine code into the top of memory and move the top of memory pointer down appropriately. An assembly code version of each program is provided for the PET. These can be easily modified to run on other 6502-based systems and, with some work, 6809-based systems as well.

The first program, **USR**, will perform an analog to digital conversion and return the converted value via the USR command. Typical usage would be:

```
10 GOSUB 60000
20 PRINT USR(1)
30 GOTO 20
```

The second program, **BUFFER**, will sample and store into a buffer 256 consecutive data points. This second routine is very useful for capturing spontaneous, non-triggerable signals. Three variables have been reserved for used with this program. They are **NEM**, **BUF** and **THRESH**. **NEM**, standing for **N**ew **E**nd of **M**emory, marks the start of the machine language routine. **BUF** is the beginning of the buffer where conversion data is stored, and **THRESH** is the address of the threshold value which must be exceeded before any data will be stored. Typical usage for this program would be:

```
10 GOSUB 6000
20 POKE THRESH,50
30 SYS NEM
40 FOR I=0 TO 256
50 PRINT PEEK(BUF+I)
60 NEXT I
70 GOTO 30
```

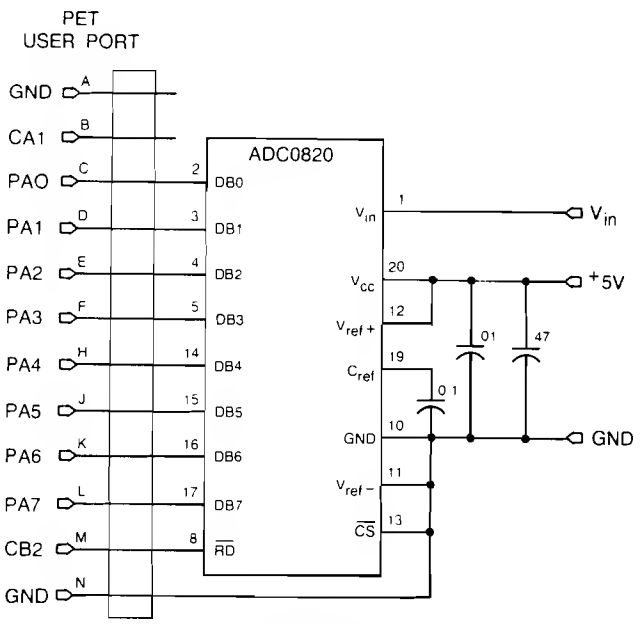
Keep in mind when using either of these programs that the top of memory will be moved down and new machine code deposited everytime the loader subroutine is called. Therefore, unless you call this loader only at the beginning of your program, you will soon be greeted with '**OUT OF MEMORY ERROR**'.

Conclusion

I hope I've shared some of my enthusiasm for this new A/D converter. We have been using them for about six months now and haven't run into a single problem. Most applications have been as data collectors in college laboratories, although I can easily imagine such applications as voice/music digitizers, solar controllers or even automotive efficiency monitors. Should anyone discover a unique use for the ADC0820, I hope you'll share your experiences with the rest of us.

Fred Genett is a Technical Associate with the Department of Chemical Engineering at University of Rochester
Department of Chemical Engineering
201 Gavett Hall; River Campus
Rochester, NY 14627

Figure 1



Note on the VIC-20, CA1-PA7 are replaced by CB1-PB7

Listing 1. USR Program: VIC Version

```
60000 : REM ** MACHINE LANGUAGE LOADER ROUTINE **
60010 : REM ** TO PERFORM AN A/D CONVERSION    **
60020 : REM ** VIA THE USR COMMAND FOR THE     **
60030 : REM **         VIC-20                  **
60040 : REM ** BY F. J. GENETT  8/1/83         **
60050 : REM ** MICRO, FEBRUARY 1984, #69       **
60060 NEM=PEEK(55)+256*PEEK(56)-28
60070 X=INT(NEM/256) : Y=NEM-256*X
60080 POKE 56,X : POKE 52,X : POKE 2,X
60090 POKE 55,Y : POKE 51,Y : POKE 1,Y : POKE 0,76
60100 FOR I=0 TO 27
60110  READ D:POKE NEM+I,D
60120 NEXT I
60130 RETURN
60140 DATA 169,0,141,18,145,173
60150 DATA 28,145,9,244,141,28
60160 DATA 145,41,223,141,28,145
60170 DATA 173,16,145,168,169,0
60180 DATA 32,145,211,96
```

Listing 2. Buffer Program: VIC Version

```
60000 : REM ** MACHINE LANGUAGE LOADER ROUTINE **
60010 : REM ** TO SAMPLE AND STORE 256         **
60020 : REM ** CONSECUTIVE DATA BYTES          **
60030 : REM ** WRITTEN FOR THE VIC-20          **
60040 : REM ** BY F. J. GENETT  8.1/83         **
60050 : REM ** MICRO, FEBRUARY 1984, #69       **
60060 NEM=PEEK(55);256*PEEK(56)-315
60070 BUF=NEM+59 : THRESH=NEM+29
60080 X=INT(NEM/256): Y=NEM-256*X
60090 POKE 56,X : POKE 52,X : POKE 55,Y : POKE 51,Y
60100 FOR I=0 TO 51
60110  READ D: POKE NEM+I,D
60120 NEXT I
60130 X=INT(BUF/256): POKE NEM+46,X: POKE NEM+45,BUF-256*X
60140 RETURN
60150 DATA 120,169,0,141,18,145
60160 DATA 170,173,28,145,9,224
60170 DATA 141,28,145,168,152,141
60180 DATA 28,145,41,223,141,28
60190 DATA 145,173,16,145,201,5
60200 DATA 144,240,152,141,28,145
60210 DATA 41,223,141,28,145,173
60220 DATA 16,145,157,0,48,232
60230 DATA 208,238,88,96
```

Listing 3. USR Program: PET Version

```
60000 : REM ** MACHINE LANGUAGE LOADER ROUTINE **
60010 : REM ** TO PERFORM AN A/D CONVERSION    **
60020 : REM ** VIA THE USR COMMAND FOR THE     **
60030 : REM **       PET 4.0 ROM               **
60040 : REM ** BY F. J. GENETT  8/1/83         **
60050 : REM ** MICRO, FEBRUARY 1984, #69       **
60060 NEM=PEEK(52)+256*PEEK(53)-28
60070 X=INT(NEM/256) : Y=NEM-256*X
60080 POKE 53,X : POKE 49,X : POKE 2,X
60090 POKE 52,Y : POKE 48,Y : POKE 1,Y : POKE 0,76
60100 FOR I=0 TO 27
60110  READ D:POKE NEM+I,D
60120 NEXT I
60130 RETURN
60140 DATA 169,0,141,67,232,173
60150 DATA 76,232,9,244,141,76
60160 DATA 232,41,223,141,76,232
60170 DATA 173,79,232,168,169,0
60180 DATA 32,188,196,96
```

Listing 4. Buffer Program: PET Version

```
60000 : REM ** MACHINE LANGUAGE LOADER ROUTINE **
60010 : REM ** TO SAMPLE AND STORE 256         **
60020 : REM ** CONSECUTIVE DATA BYTES          **
60030 : REM ** WRITTEN FOR THE VIC-20          **
60040 : REM ** BY F. J. GENETT  8.1/83         **
60050 : REM ** MICRO, FEBRUARY 1984, #69       **
60060 NEM=PEEK(52)+256*PEEK(53)-315
60070 BUF=NEM+59 : THRESH=NEM+29
60080 X=INT(NEM/256): Y=NEM-256*X
60090 POKE 53,X : POKE 49,X : POKE 52,Y : POKE 48,Y
60100 FOR I=0 TO 51
```

(continued)

```
60110  READ D: POKE NEM+I,D
60120 NEXT I
60130 X=INT(BUF/256): POKE NEM+46,X: POKE NEM+45,BUF-256*X
60140 RETURN
60150 DATA 120,169,0,141,67,232
60160 DATA 170,173,76,232,9,224
60170 DATA 141,76,232,168,152,141
60180 DATA 76,232,41,223,141,76
60190 DATA 232,173,79,232,201,5
60200 DATA 144,240,152,141,76,232
60210 DATA 41,223,141,76,232,173
60220 DATA 79,232,157,0,48,232
60230 DATA 208,238,88,96
```

Listing 5. USR Program Assembly PET Version

```
033A A9 00    INIT  LDA  #$00  ; SET DDRA FOR INPUT
033C 8D 43 E8       STA $E843
033F AD 4C E8       LDA $E84C  ; PULL CB2 HIGH
0342 09 E0          ORA #$E0
0344 8D 4C E8       STA $E84C
0347 29 DF          AND #$DF   ; PULL CB2 LOW
0349 8D 4C E8       STA $E84C
034C AD 4F E8       LDA $E84F  ; READ CONVERSION BYTE
034F A8             TAY        ; LOAD CONVERSION BYTE
0350 A9 00          LDA #$00   ; INTO FLOATING
0353 20 BC C4       JSR $C4BC  ; POINT ACCUMULATOR
0355 60             RTS        ; RETURN FROM SUBROUTINE
```

Listing 6. BUFFER Program Assembly PET Version

```
033A 78       INIT  SEI        ; DISABLE INTERRUPTS
033B A9 00          LDA #$00   ; SET DDRA FOR INPUT
033D 8D 43 E8       STA $E843
0340 AA             TAX
0341 AD 4C E8       LDA $E84C  ; PULL CB2 HIGH
0344 09 E0          ORA #$E0
0346 8D 4C E8       STA $E84C
0349 A8             TAY        ; STORE PCR IN Y REG.
034A 98       THRSH TYA        ; RECALL PCR
034B 8D 4C E8       STA $E84C  ; PULL CB2 LOW
034E 29 DF          AND #$DF
0350 8D 4C E8       STA $E84C
0353 AD 4F E8       LDA $E84F  ; READ CONVERSION BYTE
0356 C9 05          CMP #$05   ; WAIT UNTIL INPUT
0358 90 F0          BCC THRSH  ; EXCEEDS THRESHOLD
035A 98       BUFF  TYA        ; RECALL PCR
035B 8D 4C E8       STA $E84C  ; PULL CB2 LOW
035E 29 DF          AND #$DF
0360 8D 4C E8       STA $E84C
0363 AD 4F E8       LDA $E84F  ; READ CONVERSION BYTE
0366 9D 00 30       STA $3000,X  ; FILL BUFFER
0369 E8             INX        ; BUMP POINTER AND
036A D0 EE          BNE BUFF   ; EXIT IF BUFFER FULL
036C 58             CLI        ; ENABLE INTERRUPTS
036D 60             RTS        ; RETURN FROM SUBROUTINE
```

Adding Computer Senses to Your Micro

by Andrew Cornwall

Simple hardware and software permits you to add sensing circuits for pressure, balance, light, temperature and sound to almost any microcomputer.

Requirements:
Any microcomputer with at least one input line under program control.

Ed Note: Versions for the Atari, Color Computer and Apple can be found at the end of the article.

This article is an extension of notes I took while experimenting with adding simple sensing devices to the user port of my PET. In a very limited way these devices give the microcomputer senses of feeling, balance, sight, temperature, and hearing. Since the PET, VIC and C-64 have similar user port logic, the information in this article applies directly to all of these Commodore microcomputers. In fact, the parallel port is so general that very little effort is required to adapt these devices and programs to almost any microcomputer.

USER PORT FUNDAMENTALS

Figures 1a, 1b, and 1c show the location of the user port on the PET, VIC, and C-64; Figure 2 shows the pin configuration. Note that the user port pins are on the bottom of the circuit board. The pins on the top serve other functions. Connection to the user port can be made by means of a 24 pin (12 top and 12 bottom) socket, with wires soldered to the appropriate lugs. It is not advisable to attach wires directly to the printed circuit board. The individual experiments described below involve pins 'C' and 'N'.

A description of the full capabilities and utilization of the user port is beyond the scope of this article. It is sufficient here only to explain how the user port operates for parallel input.

To activate the input mode the data direction register (a special memory location) has to be POKEd with the value zero. This tells the microcomputer that all of the parallel pins of the user port are to be inputs. Although each pin may be individually programmed as either input or output depending on the POKE value, it is convenient to make all the parallel pins serve as inputs for discussing these experiments. The data direction register can be accessed at the following memory locations:

Data Direction Register

	Memory Address
PET	**59459 decimalE843 hex**
VIC	**37138 decimal9112 hex**
C-64	**56579 decimalDD03 hex**

(POKE with zero)

Once this register is set it need not be reset unless a change in input/ output mode is desired, or if the computer is turned off.

In the input mode, the user port can be thought of as a special location in memory whose value is controlled from outside the microcomputer. The

value of this memory location can be found out by a PEEK command. The user port memory locations are:

Parallel User Port
Memory Address

P E T 59457 decimalE841 hex
V I C 37136 decimal9110 hex
C - 6 4 56577 decimalDD01 hex

(Note: Each address is 2 less than the corresponding Data Direction Register Memory Address)

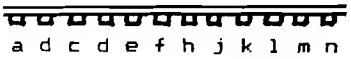
Like all memory locations there are (as many as) eight bits, with the value of each bit (zero or one) controlled by a corresponding pin on the user port. Nominally each input bit is 'high', corresponding to the voltage on the user port pin which is maintained internally by the interface adaptor chip at +5 volts. With all pins high, the value of the user port memory location is 255 decimal, FF hex, or 11111111 in binary. The way to change this value is to force the voltage on one (or more) of the pins to go somewhat below one (1) volt.

To see how this works attach wires (each about two feet long) by means of a connector to pins N and C and run the following program:

```
10  Y = 59459 :REM FOR PET
      (use: Y = 37138 FOR VIC)
      (use: Y = 56577 FOR C-64)
20  POKE (Y+2),0 :REM SET
    INPUT
30  PRINT PEEK(Y) :REM READ
    INPUT
40  GOTO 30
```

A column of '**255**' should scroll down the screen. Now touch the bare ends of the two wires together. The value on the screen should change to **254**, indicating that pin C has been shorted to pin N, which is input ground. In binary 254 is 11111110, and pin C controls the least significant bit of the parallel user port. Shorting a parallel input pin to ground does not cause any damage to the interface adaptor chip:

FIGURE 2
PIN CONFIGURATION

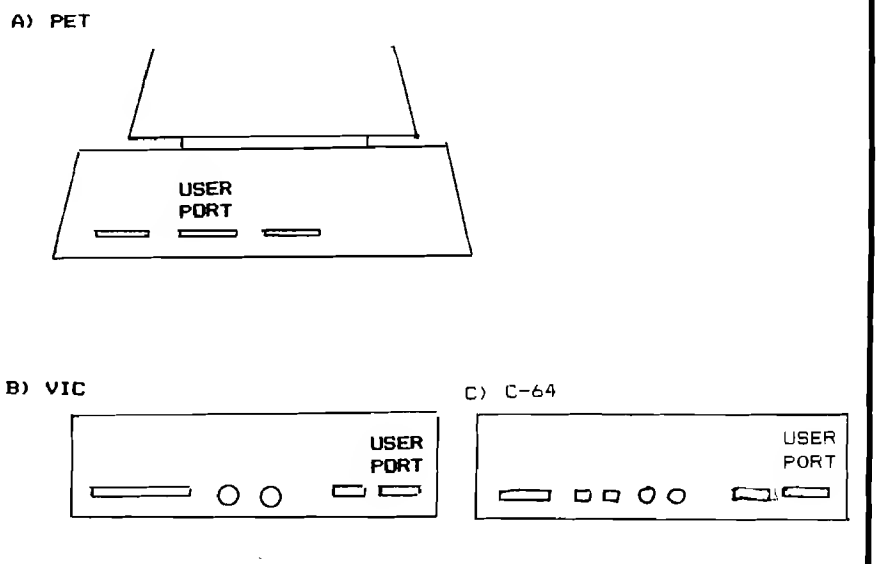


(bottom of board)

parallel pins c through l
input ground pin n (or a)

other pins:
d: ca1 on PET; cb1 on VIC
m: cb2

FIGURE 1
LOCATION OF USER PORT



this is the way it is meant to operate.

Sensing Circuits

The circuits described below are simple and inexpensive. Where possible, the parts have been identified with RADIO SHACK catalog numbers, although substitute parts with approximately the same electronic characteristics should work just as well.

A unique feature of these circuits is that they are entirely powered from the parallel user port. Maximum current from this power source is extremely small, in the order of 0.5 milliamps for the PET, 1.0 for the VIC, and .75 milliamps for the C-64.

All circuits except the 'Computer Ear' work with the same base demonstration program, listed below. Line 50 changes to provide an appropriate message for each circuit. Also, in two circuits the test condition in line 40 is subject to change from 255 to 254.

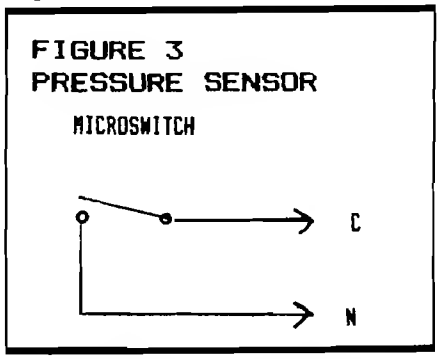
BASIC Demonstration Program

```
20  Y = 59459 :REM FOR PET
      (use: Y = 37138 FOR VIC)
      (use: Y = 56577 FOR C-64)
30  POKE (Y+2),0 :REM SET
    INPUT
40  IF PEEK (Y) = 255 THEN 40
50  REM RESERVED FOR
    MESSAGE
60  GOTO 40
```

I FEEL PRESSURE

The pressure sensing device is a microswitch (Radio Shack #275 016). This is a compact switch requiring very little force to activate. The circuit, shown in Figure 3, merely shorts pin C to pin N when the switched is closed. If

FIGURE 3
PRESSURE SENSOR



a switch has more than two lugs, those corresponding to 'normally open' and 'common' should be connected to the user port. Otherwise operation of the program will be reversed or not work at all. Program line 50 becomes:

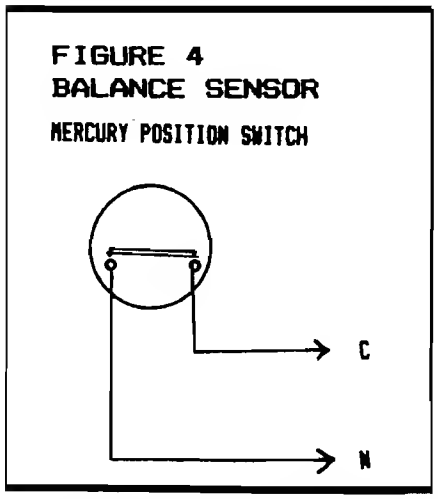
```
50  PRINT "[clr]I FEEL
    PRESSURE"
```

I'M OFF BALANCE

This is another switch activated circuit. The switch is a mercury position sensor (275 025), consisting of a small drop of mercury in a miniature case. When the switch is held level the mercury rests in the center of the case's

base, making contact between two electrodes. If the sensor is tilted, then the mercury slips off the electrodes and the connection is broken. The switch is 'on' when it is level, 'off' when tilted.

FIGURE 4
BALANCE SENSOR



A circuit diagram for the position sensor is shown in Figure 4. Program lines 40 and 50 are:

```
40 IF PEEK (Y) = 254 THEN 40
50 PRINT "[clr]I'M OFF
   BALANCE"
```

The test condition in line 40 is changed to reflect the normally closed condition of the position sensor when level. I found that my sensor was very sensitive, only indicating level within a narrow range of positions.

I SEE LIGHT

This circuit is a little more complicated than the previous two. It uses a phototransistor (276 130) as a light activated variable resistor. A 5K (or 5,000) ohm variable resistor (or potentiometer or trim pot) is added for sensitivity control.

The resistance of the phototransistor becomes lower as more light enters through the top of its case. When connected to the user port, the phototransistor will cause the interface adaptor chip's pin voltage to decrease as the amount of light increases. With sufficient light, pin voltage will drop below the low-condition threshold.

The addition of a variable resistor allows for sensitivity adjustment. There are two ways to connect the variable resistor, shown in Figures 5A and 5B, one for low and the other for high level light conditions. In either case, I found the light detection capability to be quite good, able to respond to small changes in lighting, such as a shadow cast over the phototransistor. Program line 50 is:

```
50 PRINT "[clr]I SEE LIGHT"
```

FIGURE 5
LIGHT SENSOR

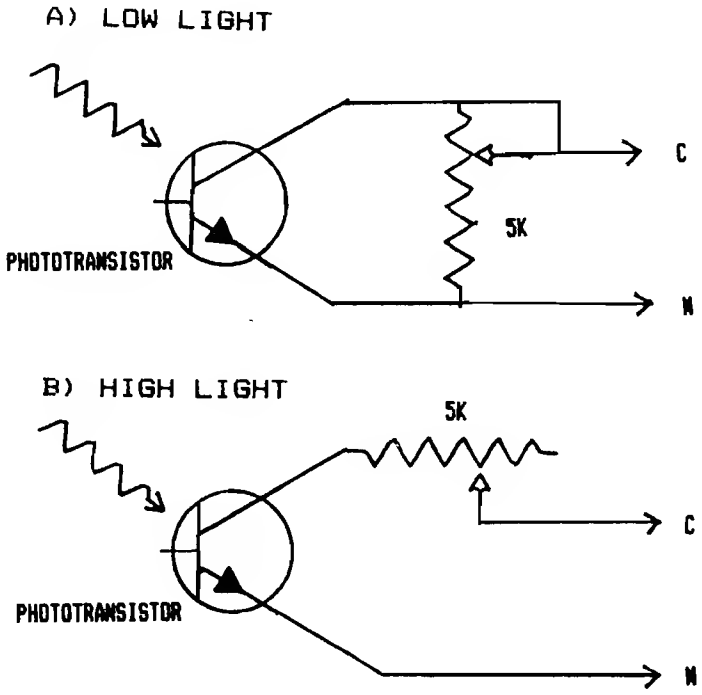
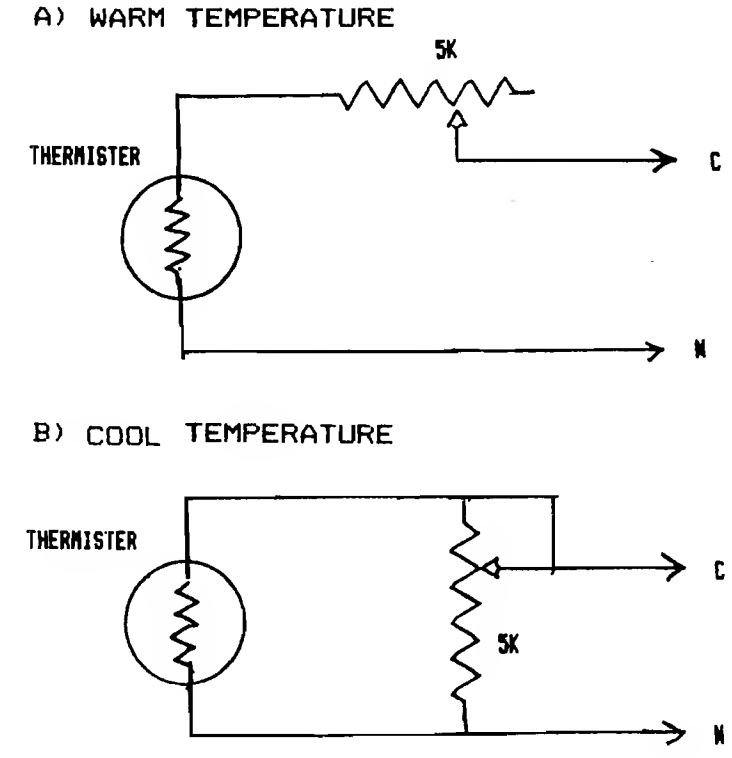


FIGURE 6
TEMPERATURE SENSOR



I'M GETTING WARM

This circuit also relies on a sensing device that changes resistance. In this instance the device is a thermister which responds to changes in temperature. The thermister's resistance decreases with rising temperature. Thermisters are available in a wide variety of types and characteristics. I used a junk box thermister similar to a Fenwal JA35J1, which is a small wafer thermister having resistance of 5k (5,000) ohms at 25 degrees celsius (77 degrees Fahrenheit).

When connected to the user port, a normally 'cool' thermister should cause the pin voltage to be high. As the temperature increases (e.g., warming it with your fingers) pin voltage will drop below the low-threshold value.

The circuits for the temperature sensor are shown in Figures 6A and 6B. Again, there are two options for wiring the 5k ohm variable resistor as a sensitivity control, one for high temperature and another for low temperature operating range. Program lines 40 and 50 become:

```
40 IF PEEK(Y) = 255 THEN 40
50 PRINT "[clr]I'M GETTING
   WARM"
```

When working with a thermister for critical temperature applications, let the thermister operate for several minutes before making the final sensitivity adjustment.

I HEAR A NOISE

This circuit senses the presence of audio energy. It is not sophisticated enough to interpret sound, but it will tell if a sound is being made within the range of 'hearing' of the computer. I like to refer to this circuit as the "Computer Ear".

The Ear's circuitry, as shown in Figure 7, consists of a crystal microphone (270-095), two transistors arranged as a Darlington pair, and a 5k ohm potentiometer. The sensing device is a microphone which generates a small voltage in relation to the sound energy it receives. This circuit may be thought of as working in a way similar to the light and temperature detectors, aabove. The transistors are needed to amplify the small voltages produced by the microphone and to convert these into a change in resistance as seen by the user port. The potentiometer provides for sensitivity control. In addition to these parts, there is bias resistor which seemed to work equally well in range of 1 to 10 megohms (1 million to 10 million ohms), and a small capacitor to suppress a tendency for the circuit to oscillate with the user port.

Detecting sound can be tricky since sound energy is in waves and sampling of the user port is periodic and nearly instantaneous. For a sound to be heard, a PEEK has to occur at the same time that a sound wave strikes the microphone with enough energy to affect the user port.

To obtain better sound capture I mounted the microphone at the inner apex of a six-inch heavy paper cone. This improved the sensitivity of the microphone, but made it more directional. Another equipment consideration was to use transistors (general purpose, small signal, NPN type) with fairly high beta (or gain) factor (about 150 Hfe).

The real improvement in sensitivity, however, occurs from using a WAIT statement to PEEK the user port. I am grateful to Ken Beverley of the Nova Scotia Commodore Computer Users Group for advising me to use this command. The advantage of this statement is that it examines the user port at a very rapid rate, thereby increasing the chance of sampling at the high point of a sound wave. The following program shows the use of the WAIT statement for the Computer Ear.

```
20 Y = 59459 :REM FOR PET
     (use: Y = 37138 FOR VIC)
     (use: Y = 56577 FOR C-64)
30 POKE (Y+2),0 :REM SET
   INPUT
35 WAIT Y,1,255
40 PRINT"I HEAR A NOISE"
50 FOR J=1 TO 1000: NEXT J
60 PRINT [clr]
70 GOTO 35
```

In this program the WAIT statement will continue rapid examination of the user port so long as the value of the port remains at 255. As soon as there is a noise the port value goes to 254, the program is allowed to continue, and the message will be printed temporarily.

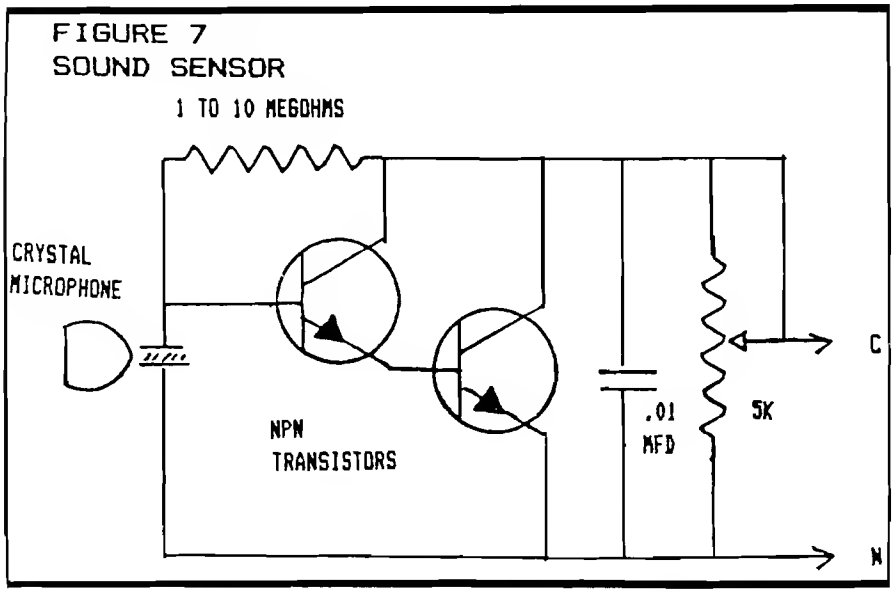
The sensitivity of the computer ear will vary from one microcomputer to another depending on the personality of its input port chip. I have found the computer ear to be reasonably sensitive on my PET, less sensitive on my son's C-64, but remarkably sensitive on another PET and a friend's VIC. Chances are that it will work fairly well on most PETs, VICs, and C-64s.

CARRY ON

These are demonstration circuits, but they can be easily extended to use a PET, VIC, or C-64 to monitor a variety of sensory conditions. More than one device can be connected to the user port at one time by attaching each to a different parallel input pin. In most cases, change in status of an individual sensor can be detected by means of AND statements to decode the parallel port value for each pin.

Adding computer senses to other computers involves essentially the same steps as those for the Commodore computers. The major differences lie in the location and/or characteristics of the parallel I/O device. The information given here was obtained by examining the ATARI and CoCo manuals. I have NOT had a chance to test them, so consider it *reference*

FIGURE 7
SOUND SENSOR



material that should at least get you going in the right direction, and might (Murphy's Laws aside) even work!

ATARI

The Atari has a PIA located at addresses $D300 to $D31F. This is a **6520**, which while similar to the **6522** used in the Commodore, has a few differences. The direction of each bit in the DATA register (input or output) is set in a manner quite different from the method of the 6522. Instead of having an entire register address dedicated to the direction register, as in the 6522, a single bit in a CONTROL register changes the meaning of the register at $D300 from DATA to DIRECTION information. If bit 2 ($04) in $D302 is set to 0, then the register at $D300 is open for modification as the DIRECTION register. Bits in $D300 set to 0 become input, bits set to 1 become output. No input or output is possible at this time. When bit 2 at $D302 is set to 1, then the register at $D300 becomes a normal DATA register. The BASIC Demonstration program becomes:

This I/O device is connected to the Atari Controller Jack number 1 as data bit 0. You should be able to connect the experimental devices to this location. Connect **C** to pin 1 of controller jack 1. Connect **N** to pin 8 of the controller jack 1.

Note: Lines 30 to 36 may not be required since the Atari normally is expecting input from the joysticks and has this port bit set for input.

Color Computer

The technique for the CoCo is very similar to that for the ATARI presented above. It uses a **6821 PIA**. This has its DATA Register at $FF20 and a CONTROL Register at $FF21. Bit 2 of this CONTROL Register controls the functioning of the DATA Register. If bit 2 is equal 0, then the DATA Register functions as a DATA DIRECTION Register. Each bit in $FF20 may then be individually set (output) or cleared (input). When bit 2 is equal to 1, then the DATA Register functions as an I/O port. Bit 0 of this register is connected to the Cassette Data Input, and my be used for the computer senses.

Note: The steps in lines 30 to 34 may not be required since the CoCo usually has bit 0 of the PIA at $FF20 set as input for the cassette.

The **C** connections from the experiments should be to pin 4 of the cassette interface connector. The **N** ground should be to pin 2.

MICRO™

My thanks to Kevin Stone of the Nova Scotia Commodore Users Group, and to my son, Andrew, for reviewing this article and trying out the programs on their respective VIC and C-64 microcomputers.

Andrew Cornwall may be reached at:
66 Landrace Cresent
Dartmouth, Nova Scotia
B2W 2P9 Canada

```
atari
   20 Y = 54016 :
      REM = $D300 PIA ADDRESS
   30 POKE (Y+2),(PEEK(Y+2)-4) :
      REM SET DIRECTION REGISTER
   32 POKE (Y),0 :
      REM SET INPUT ON JOYSTICK 1
   34 POKE (Y+2),(PEEK(Y+2)+4) :
      REM SET DATA REGISTER
   40 IF PEEK (Y) = 255 THEN 40
   50 REM RESERVED FOR MESSAGE
   60 GOTO 40
```

```
coco
   20 Y = 65312 :
      REM = $FF20 PIA ADDRESS
   30 POKE (Y+1),(PEEK(Y+1)-4) :
      REM SET DIRECTION REGISTER
   32 POKE (Y),254 :
      REM SET INPUT ON CASSETTE DATA
   34 POKE (Y+1),(PEEK(Y+1)+4 :
      REM SET DATA REGISTER
   40 IF PEEK (Y) = 255 THEN 40
   50 REM RESERVED FOR MESSAGE
   60 GOTO 40
```

DOSPLUS for Commodore 64

Part 2

by Michael Keryan

DOSPLUS-- Add more new utility functions to a Commodore 64 by use of the RESTORE key.

This article adds new functions to the recently published DOSPLUS utility program. New functions generate a system HELP screen, a USER HELP screen, and do number conversions. In addition, a transient program relocator is provided that will allow quite a few other utilities to occupy the same address space: $C000-C7FF.

Listing 1

```
; ADDITIONAL UTILITIES FOR DOSPLUS
;
;       M. J. KERYAN
;       12-06-83
;
;       USE WITH DOSPLUS FROM
;       MICRO -- 1984 --
;
;TRANSIENT PROGRAM VECTOR
;PULLS IN ROUTINES FROM UNDER
;BASIC ROM IF NOT CURRENTLY IN RAM
```

In the previous article (**DOSPLUS** in **MICRO #68, January 1984**), seven machine language utility functions were provided: change background, border, and character colors; turn on and off a printer (device 4); dump the screen to the printer; and append programs. All DOSPLUS routines can be called by using the RESTORE key (which generates a non-maskable interrupt) so that you can use them at any time--in BASIC immediate mode, in SUPERMON, or even in the midst of a running BASIC program. The framework was provided to allow a total of 32 such functions to be added. DOSPLUS was assembled to sit alongside the DOS 5.1 wedge program, so they could be loaded and initialized together.

The Commodore 64 contains 64K bytes of progammable memory (RAM), but our BASIC programs only have about 38K of free memory space. Why? About 1K ($0000-03FF) is used for system support: pointers, stack, input buffers, etc. About 1K ($0400-07FF) is used to store the character codes for the screen. 8K of memory space ($A000-BFFF) is tied up by the BASIC ROM and 8K more ($E000-FFFF) by the KERNAL ROM. Another 4K ($D000-DFFF) is assigned to various Input/Output peripheral chips, the character generator, and the color memory. This leaves 38K memory at $0800-9FFF, usable by BASIC, and another 4K at $C000-CFFF, that is free memory but not usable for BASIC program or variable storage. This 4K memory bank is used by most Commodore 64 machine language programs including the DOS wedge and DOSPLUS.

Although it isn't exactly easy, the 'hidden' RAM underneath the BASIC and KERNAL ROMs is usable by self-sufficient machine language programs. The Commodore 64's 6510 CPU contains a memory management register (addressed at $0001) that can be set to switch between various RAM or ROM configurations. When the ROMs are switched out, however, care must be taken to ensure that no ROM utilities are called; this includes interrupts. In this installment of DOSPLUS, we use the 'hidden' RAM in two ways. Screen data is stored in the KERNAL area, and additional utility programs are stored in the BASIC area.

```
                        ;ALL SUCH ROUTINES EXECUTE AT
                        ;THE SAME LOCATION -- $C000
                        ;
                        R6510  = $01
                        TBLOCK = $A000      ;TABLES UNDER
                        TSTART = $A020      ;BASIC ROM
                        ;
                        *      = $CF59      ;UNUSED SPACE
                        ;                   IN DOS 5.1
                        ;
CF59  EC 83 CF          TRNSLD CPX TRNKEY   ;KEY IN RAM?
CF5C  F0 22                    BEQ TRANSF   ;YES
CF5E  8E 83 CF                 STX TRNKEY   ;NO, SWITCH
CF61  78                       SEI
CF62  A5 01                    LDA R6510    ; OUT BASIC
CF64  29 FE                    AND #$FE     ; ROM
CF66  85 01                    STA R6510
CF68  BD 20 A0                 LDA TSTART,X ;ROUT. LOCAT.
CF6B  8D 88 CF                 STA FROM+2   ;SET UP SUBR.
CF6E  BC 00 A0                 LDY TBLOCK,X ;# OF BLOCKS
CF71  A9 C0                    LDA #$C0
CF73  8D 88 CF                 STA TO+2
CF76  20 84 CF                 JSR MOVE     ;MOVE IT
CF79  A5 01                    LDA R6510    ;SWITCH BASIC
CF7B  09 01                    ORA #$01     ; ROM BACK IN
CF7D  85 01                    STA R6510
CF7F  58                       CLI
CF80  4C 00 C0          TRANSF JMP $C000    ;EXECUTE IT
CF83  00                TRNKEY .BYTE $00    ;CURRENT KEY
                        ;
                        ;SUBROUTINE TO MOVE MEMORY BLOCKS
                        ;SET UP Y=# OF 256 BYTE BLOCKS
                        ;TRANSFER, FROM+2 AND TO+2 TO
                        ;CORRESPONDING BLOCK #'S
                        ;
CF84  A2 00             MOVE   LDX #$00
CF86  BD 00 A1          FROM   LDA $A100,X  ;MODIFIABLE
CF89  9D 00 C0          TO     STA $C000,X  ; LOCATIONS
CF8C  E8                      INX
CF8D  D0 F7                   BNE FROM
CF8F  EE 88 CF                INC FROM+2
CF92  EE 8B CF                INC TO+2
CF95  88                      DEY
CF96  D0 EE                   BNE FROM
CF98  60                      RTS
                        ;
                        ;SCREEN TRANSFER ROUTINES
                        ;FOR ALPHA. SCREEN AT $0400
                        ;                  TO $F000
                        ;AND COLOR SCREEN AT $D800
                        ;                  TO $F400
                        ;
CF99  A2 00             SCRSAV LDX #$00     ;VIDEO MATRIX
CF9B  20 A0 CF                 JSR LOADTR
CF9E  A2 01                    LDX #$01     ;COLOR MEM.
CFA0  BD 10 C8          LOADTR LDA SCRTB1,X ;SUBR. TO SET
CFA3  8D 88 CF                 STA FROM+2   ; UP MOVE SUBR.
CFA6  BD 18 C8                 LDA SCRTB2,X ;
CFA9  8D 8B CF                 STA TO+2
CFAC  A0 04                    LDY #$04     ;# OF BLOCKS
CFAE  4C 84 CF                 JMP MOVE
                        ;
CFB1  78                SCRRCL SEI          ;DISABLE IRQ
```

Transient Program Relocator

Quite a few Commodore 64 machine language utility programs are available in the public domain. They are generally well-written, self-contained programs that almost always are written to execute at $C000, although some have multiple entry points above this address. Rather than trying to re-invent the wheel and attempt to sqeeze all their functionality into the $C000-C7FF memory space left after DOS 5.1 and DOSPLUS are loaded, we will now do the seemingly impossible: allow all these programs to be used with DOSPLUS and have them all resident at $C000.

This is done with a transient program relocator, as shown in the assembly source code (Listing 1). The TRNSLD routine will move a program located in the hidden RAM area of $A000-BFFF to a workable location of $C000. All these hidden programs will have their table pointers (located in tables TABL and TABH from last time) pointing to TRNSLD. If the desired transient program is not the one currently in the $C000 memory, new tables located in the hidden RAM will tell the routine how large the new program is (in 256 byte blocks) and where the new program is residing. These table are stored at $A000-A01F and $A020-A03F. The hidden program is then transferred to $C000+ and execution is transferred to $C000.

To make things simple, some restrictions are placed on these transient programs.

- They must be assembled to run at $C000.
- They are located at the beginning of a block, e. g. $A100 or $BC00.
- The first block of the hidden RAM ($A000-A0FF) is used by tables and can't be used for programs.
- Only one such program can be resident in the working area at any one time, so they must be independent (or have the ability to call others into the working area gracefully).

The MOVE subroutine is a general purpose memory relocator used by several routines described later, as well as TRNSLD. This routine will only move whole blocks of memory (256 bytes each). More details of the transient programs and some tricks in using them will be given in a future article. Before we have to use the hidden RAM for program storage, let's

```
CFB2  A5 01           LDA R6510    ;TO RECALL,
CFB4  29 FD           AND #$FD     ; SWITCH OUT
CFB6  85 01           STA R6510    ; KERNAL ROM
CFB8  A2 02           LDX #$02
CFBA  20 A0 CF        JSR LOADTR   ;RECALL VIDEO
CFBD  A2 03           LDX #$03
CFBF  20 A0 CF        JSR LOADTR   ;RECALL COLOR
CFC2  A5 01     ROMIN LDA R6510    ;SWITCH BACK
CFC4  09 02           ORA #$02     ; KERNAL ROM
CFC6  85 01           STA R6510
CFC8  58              CLI          ;ENABLE IRQ
CFC9  60              RTS
                ;
CFCA  78        WRTHLP SEI         ;DISABLE IRQ
CFCB  A5 01           LDA R6510    ;HELP SCREEN
CFCD  29 FD           AND #$FD     ; IS ALSO
CFCF  85 01           STA R6510    ; UNDER KERNAL
CFD1  A2 04     X4    LDX #$04
CFD3  20 A0 CF        JSR LOADTR
CFD6  A2 05     X5    LDX #$05
CFD8  20 A0 CF        JSR LOADTR
CFDB  20 E1 CF        JSR COLOR    ;SWITCH COLOR
CFDE  18              CLC
CFDF  90 E1           BCC ROMIN    ;KERNAL BACK
                ;
                BRDCLR = $D020     ;BORDER COLOR
                BCKCL1 = $D021     ;BACKGROUND
                ;
                ;COLORS OF HELP SCREENS ARE STORED
                ;AT BYTES 1001 AND 1002 OF THE
                ;1024 BYTE VIDEO MATRIX FOR THE
                ;BORDER AND BACKGROUND
                ;
CFE1  A2 04     COLOR LDX #$04
CFE3  BC 10 C8        LDY SCRTB1,X
CFE6  C8              INY
CFE7  C8              INY
CFE8  C8              INY
CFE9  8C F3 CF        STY CL1+2    ;SET UP
CFEC  8C FA CF        STY CL2+2
CFEF  A2 E8           LDX #$E8     ;HI+3*256
CFF1  BD 00 FB  CL1   LDA $FB00,X  ; +E8(HEX)
CFF4  8D 20 D0        STA BRDCLR
CFF7  E8              INX
CFF8  BD 00 FB  CL2   LDA $FB00,X  ; +1 MORE
CFFB  8D 21 D0        STA BCKCL1
CFFE  60              RTS
                ;
C810            *     = $C810
                ;
C810  04 D8 F0 F4   SCRTB1 .BYTE $04,$D8,$F0,$F4
C814  F8 FC E8 EC          .BYTE $F8,$FC,$E8,$EC
C818  F0 F4 04 D8   SCRTB2 .BYTE $F0,$F4,$04,$D8
C81C  04 D8 04 D8          .BYTE $04,$D8,$04,$D8
                  GETIN  = $FFE4
                  ;
                  ;HELP SAVES CURRENT SCREEN AND
                  ;COLORS, WRITES THE HELP SCREEN,
                  ;WAITS FOR CARRIAGE RETURN,
                  ;THEN RESTORES EVERYTHING
                  ;
C820  AD 20 D0    HELP  LDA BRDCLR   ;SAVE CURRENT
C823  48                PHA          ;COLORS
```

```
C824  AD 21 D0            LDA BCKCL1
C827  48                  PHA
C828  20 99 CF            JSR SCRSAV    ;SAVE SCREEN
C82B  20 CA CF            JSR WRTHLP    ;HELP SCREEN
C82E  20 E4 FF     WAITCR JSR GETIN     ;WAIT KEYBOARD
C831  C9 0D               CMP #$0D      ;CARR RETURN?
C833  D0 F9               BNE WAITCR
C835  20 B1 CF            JSR SCRRCL
                         ;RESTORE SCREEN
C838  68                  PLA           ; AND COLORS
C839  8D 21 D0            STA BCKCL1
C83C  68                  PLA
C83D  8D 20 D0            STA BRDCLR
C840  60                  RTS
                  ;
C841  A0 06       USRHLP  LDY #$06      ;USES THE HELP
C843  8C D2 CF            STY X4+1      ; ROUTINE
C846  8C E2 CF            STY COLOR+1   ; WITH USR
C849  C8                  INY           ; SCREEN AT
C84A  8C D7 CF            STY X5+1      ; $E800-
C84D  20 20 C8            JSR HELP
C850  A0 04               LDY #$04      ;RESTORE
C852  8C D2 CF            STY X4+1      ; COLORS
C855  8C E2 CF            STY COLOR+1
C858  C8                  INY
C859  8C D7 CF            STY X5+1
C85C  60                  RTS
                  ;
                  *       = $CB60
                  ;
                  ;NUMBER CONVERSION ROUTINES
                  ;HEX/DEC AND DEC/HEX
                  ;
                  NUML    = $FD         ;SHARE THESE
                  NUMH    = $FE         ; WITH DUMP
                  NUMTMP  = $02FF
                  CHRIN   = $FFCF
                  CHROUT  = $FFD2
                  MESSAG  = $CB41
                  ;
CB60  20 99 CF    NUMBER  JSR SCRSAV    ;SAVE SCREEN
CB63  A9 00       NUMCTD  LDA #$00      ;SET NUMBER TO
CB65  85 FD               STA NUML      ; ZERO
CB67  85 FE               STA NUMH
CB69  20 41 CB            JSR MESSAG
CB6C  0D 4E 55 4D         .BYTE $0D,$4E,$55,$4D
CB70  42 45 52 20         .BYTE $42,$45,$52,$20
CB74  3F 20 00            .BYTE $3F,$20,$00
CB77  20 CF FF            JSR CHRIN     ;GET INPUT
CB7A  C9 0D               CMP #$0D
CB7C  F0 3C               BEQ NUMRTN    ;RETURN = END
CB7E  C9 24               CMP #$24      ;$ FOR HEX NO.
CB80  D0 03               BNE DCIMAL
CB82  4C 60 C8            JMP HEX
CB85  C9 30       DCIMAL  CMP #$30      ;<ZERO?
CB87  90 DA               BCC NUMCTD    ;YES, GO BACK
CB89  C9 3A               CMP #$3A      ;>9?
CB8B  B0 D6               BCS NUMCTD    ;YES, GO BACK
CB8D  29 0F               AND #$0F      ;CONVERT BIN
CB8F  A2 11               LDX #$11
CB91  D0 05               BNE D3
CB93  90 02       D1      BCC D2
CB95  69 09               ADC #$09
CB97  4A          D2      LSR A
```

fill up the free holes in the $C800-CFFF memory area.

Help Screens

Help screens are great. They allow you to call for instructions or options without having to refer to manuals or overlays. They are especially helpful when certain keys are defined to execute different functions as in the DOS wedge or DOSPLUS. DOSPLUS uses the hidden RAM under the KERNAL ROM to store screens of data. DOSPLUS supports two help screens that can be called at any time by RESTORE, H (or SYS 51232) or RESTORE, U (or SYS 51265).

When a help screen is called, the current screen is first saved. Character memory ($0400-07FF) is transferred to hidden RAM (at $F000-F3FF) and color memory ($D800-DBFF) is also moved (to $F400-F7FF). The HELP and USRHLP routines also save the current border and background color codes.

After the screen data is safely tucked away, hidden help screens are transferred to the screen. The HELP character memory resides at $F800-FBFF (USRHLP at $E800-EBFF) and the color memory is found at $FC00-FFFF ($EC00-EFFF for USRHLP). The border and background colors are stored in the 1001th and 1002th byte of the matrix (the last 22 bytes are unused).

Once the help screen is displayed, the keyboard is scanned until a RETURN is detected. At that point the old screen is restored and everything is back to normal. The simplest way to load the help screens into hidden RAM is illustrated in the boot program DOSPLUS+ (Listing 3). Print statements are used to place the desired information on the screen (using color and other control characters if desired). A few pokes are made to set up the MOVE routine and then the fast machine language MOVE program is called by the SYS command to transfer the screen.

Number Conversions

Number conversions are a pain. Basic uses decimal numbers 0-65535 or 0-255 for all of its PEEKs and POKEs. Machine language monitor programs use hexadecimal numbers $0-FFFF. The number conversion routines in DOSPLUS will convert from decimal to hex and from hex to decimal. Since the

routines can be called at any time, they can be used while in machine language monitor programs as well as while in BASIC.

Press RESTORE then N. Although not obvious at this time, the current screen will then be saved (and later restored). You will then be prompted to enter a number. If you enter a number from 0 to 65535, the corresponding hexadecimal number will be displayed. To convert a hexadecimal number, enter it with a leading dollar sign, e. g. $3D or $FF00. Don't try to enter a number greater than 65535 (or $FFFF), but numbers with fewer digits are just fine.

After the number is converted, you will then get another prompt. You can then continue to enter numbers, or enter just a RETURN to end. As with most DOSPLUS routines, you can also call the number conversion routine inside a BASIC program with a SYS. As an exercise, use the number conversion routine to calculate the corresponding decimal number to use with SYS.

Entering the Program

Assuming you have the original DOSPLUS routines in memory, enter the new code either with an assembler (using Listing 1) or through the BASIC loader (Listing 2). A third method is to send the author $10 (US) for a disk containing the original and additional utility programs. (For foreign requests, please send sufficient return postage.)

The BASIC program DOSPLUS+ (Listing 3) will boot all the DOSPLUS routines as well as the DOS WEDGE. It also initializes both utilities and stores the help screens under the KERNALROM. It then NEWs itself and prints the available free memory.

You can change DOSPLUS to also boot your own BASIC program with its own customized USRHLP screen. Replace the PRINT statement of line 1510 to a number of PRINT statements that set up your customized screen. Delete line 3060 which is used only to display the HELP screen on boot-up. Then change the following lines:

```
3020 PRINT "{HOME}LOAD"+CHR$(34)+
     "PGMNM"+CHR$(34)+",8"
3090 POKE 198,6
3100 POKE 631,19:POKE 632,13:POKE 633,82
3105 POKE 634,85:POKE 635,78:POKE 636,13
```

PGMNM is the name of your program that you want to boot. Lastly, save the revised DOSPLUS+ program under a new name that is associated with the final program to be booted.

```
CB98  66 FE        D3     ROR NUMH
CB9A  66 FD               ROR NUML
CB9C  CA                  DEX
CB9D  D0 F4               BNE D1
CB9F  20 CF FF            JSR CHRIN
CBA2  C9 0D               CMP #$0D      ;CAR RETURN?
CBA4  D0 DF               BNE DCIMAL
CBA6  20 41 CB            JSR MESSAG    ;PRINT MESSAGE
CBA9  20 20 24 00         .BYTE $20,$20,$24,$00
CBAD  A5 FE               LDA NUMH      ;WRITE HEX
CBAF  20 CB C8            JSR WRTHX     ; EQUIVALENT
CBB2  A5 FD               LDA NUML
CBB4  20 CB C8            JSR WRTHX
CBB7  4C 63 CB            JMP NUMCTD    ;CONTINUE
                          ;
CBBA  4C B1 CF    NUMRTN  JMP SCRRCL    ;RESTORE SCREEN
                          ;
                          *       = $C85D
                          ;
C85D  4C 63 CB    MISTAK  JMP NUMCTD    ;BACK TO INPUT
C860  20 CF FF    HEX     JSR CHRIN     ;INPUT IS HEX
C863  C9 0D               CMP #$0D
C865  F0 22               BEQ WRTDEC    ;DONE WITH NO.
C867  C9 30               CMP #$30      ;<ZERO?
C869  90 F2               BCC MISTAK
C86B  C9 3A               CMP #$3A      ;>9?
C86D  90 0A               BCC H1
C86F  C9 41               CMP #$41      ;<A?
C871  90 EA               BCC MISTAK
C873  C9 47               CMP #$47      ;>F?
C875  B0 E6               BCS MISTAK
C877  E9 36               SBC #$36      ;CONVERT BIN
C879  0A          H1      ASL A
C87A  0A                  ASL A
C87B  0A                  ASL A
C87C  0A                  ASL A
C87D  A2 04               LDX #$04
C87F  0A          H2      ASL A
C880  26 FD               ROL NUML
C882  26 FE               ROL NUMH
C884  CA                  DEX
C885  D0 F8               BNE H2
C887  F0 D7               BEQ HEX
                          ;
C889  20 41 CB    WRTDEC  JSR MESSAG    ;PRINT MESSAGE
C88C  20 20 00            .BYTE $20,$20,$00
C88F  A0 07       D4      LDY #$07      ;CONVERT TO
C891  A2 30       D5      LDX #$30      ; DECIMAL
C893  38          D6      SEC           ; EQUIVALENT
C894  A5 FD               LDA NUML
C896  F9 C2 C8            SBC TABCON-1,Y
C899  48                  PHA
C89A  88                  DEY
C89B  A5 FE               LDA NUMH
C89D  F9 C4 C8            SBC TABCON+1,Y
C8A0  90 09               BCC D7
C8A2  85 FE               STA NUMH
C8A4  68                  PLA
C8A5  85 FD               STA NUML
C8A7  E8                  INX
C8A8  C8                  INY
C8A9  D0 E8               BNE D6
C8AB  68          D7      PLA
C8AC  8A                  TXA
```

```
C8AD  8C FF 02            STY NUMTMP
C8B0  20 D2 FF            JSR CHROUT
C8B3  AC FF 02            LDY NUMTMP
C8B6  88                  DEY
C8B7  10 D8               BPL D5
C8B9  A5 FD               LDA NUML
C8BB  09 30               ORA #$30
C8BD  20 D2 FF            JSR CHROUT
C8C0  4C 63 CB            JMP NUMCTD
                    ;
C8C3  0A 00         TABCON .BYTE $0A,$00  ;10
C8C5  64 00                .BYTE $64,$00  ;100
C8C7  E8 03                .BYTE $E8,$03  ;1000
C8C9  10 27                .BYTE $10,$27  ;10000
                    ;
C8CB  48            WRTHX  PHA            ;WRITE HEX NO.
C8CC  4A                   LSR A
C8CD  4A                   LSR A
C8CE  4A                   LSR A
C8CF  4A                   LSR A
C8D0  20 D6 C8             JSR W1
C8D3  68                   PLA
C8D4  29 0F                AND #$0F
C8D6  C9 0A         W1     CMP #$0A
C8D8  B0 04                BCS CONASC
C8DA  09 30                ORA #$30
C8DC  D0 02                BNE WRTBYT
C8DE  69 36         CONASC ADC #$36
C8E0  4C D2 FF      WRTBYT JMP CHROUT
                    ;
C8E3  20 41 CB      COPMES JSR MESSAG
                    ;(UP ARROW) ROUTINE
C8E6  20 44 4F 53          .BYTE $20,$44,$4F,$53
C8EA  50 4C 55 53          .BYTE $50,$4C,$55,$53
C8EE  20 43 2E 31          .BYTE $20,$43,$2E,$31
C8F2  39 38 34 20          .BYTE $39,$38,$34,$20
C8F6  4D 4B 45 52          .BYTE $4D,$4B,$45,$52
C8FA  59 41 4E 0D 00       .BYTE $59,$41,$4E,$0D,$00
C8FF  60                   RTS
                    ;
                    ;CHANGE POINTERS IN ENTRY TABLE
                    ;
                    *      = $CBC8       ;HELP  = $C820
CBC8  20                   .BYTE $20
                    *      = $CBD5       ;USRHLP= $C841
CBD5  41                   .BYTE $41
                    *      = $CBCE       ;NUMBER= $CB60
CBCE  60                   .BYTE $60
                    *      = $CBDE
                           ;(UP ARROW)MESSG= $C8E3
CBDE  E3                   .BYTE $E3
                    *      = $CBE8
CBE8  C8                   .BYTE $C8
                    *      = $CBF5
CBF5  C8                   .BYTE $C8
                    *      = $CBEE
CBEE  CB                   .BYTE $CB
                    *      = $CBFE
CBFE  C8                   .BYTE $C8
                    ;
CBFF                       .END
```

(article continues on page 47)

COMPUTER MAIL ORDER

TERMINALS

914 ..... $569.00
924 ..... $689.00
925 ..... $739.00
950 ..... $929.00
970 ..... $1039.00

COMPUTERS

Teleport Portable ..... CALL
800A $1099.00
802 ..... $2699.00
803 ..... $1949.00
802H $4695.00
806/20 ..... $4999.00
816/40 ..... $9199.00
1602 ..... $3399.00
1603 ..... CALL

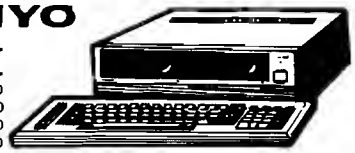
TeleVideo

Spirit XL Portable $3999.00
IIE-1 ..... $1569.00
IIE-2 ..... $1899.00
IIE-3 ..... $2399.00
IIE-4 ..... $3299.00
PC-E $1579.00
PC-1 ..... $2399.00
PC-2 ..... $2799.00
PC-XL $3599.00
1620 ..... $3599.00
1630 ..... $5499.00
1640 ..... $6499.00
Cyma Software ..... CALL

SANYO

MBC-550 PC ... CALL
MBC-555 PC ... CALL

MBC 1100 ..... $1499.00
FDD 3200-320K Drive $389.00
MBC 1150 ..... $1899.00
MBC 1200 ..... $1849.00
FDD 6400-640K Drive $469.00
M8C 1250 ..... $2099.00
PR 5500 Printer $599.00



MONITORS

AMDEK

300 Green $149.00
300 Amber $159.00
310 Amber $169.00
Color 1 ..... $279.00
Color 1 Plus $299.00
Color 2 ..... $399.00
Color 2 Plus $419.00
Color 3 ..... $349.00
Color 4 ..... $699.00

BMC

12" Green $79.99
12" Green HI-RES $119.99
9191-13" Color $249.99

GORILLA

12" Green $88.99
12" Amber $95.99

NEC

JB 1260 Green $109.99
JB 1201 Green $149.99
JB 1205 Amber $159.99
JC 1215 Color $299.99
JC 1216 RGB $429.99

PRINCETON GRAPHICS

HX-12 RGB $519.00

TAXAN

12" Green $119.00
12" Amber $129.00
Taxan 1 RGB $279.00

USI

Pi 1, 9" Green $99.99
Pi 2, 12" Green $119.99
Pi 3, 12" Amber $149.99
Pi 4, 9" Amber $139.99
1400 Color $269.99

ZENITH

ZVM 122 Amber $109.99
ZVM 123 Green $89.99
ZVM 135 Color/RGB $469.99

MODEMS

ANCHOR

Mark I (RS-232) $79.00
Mark II (Atari) $79.00
Mark III (TI-99) $109.00
Mark IV (CBM/PET) $125.00
Mark V (Osborne) $95.00
Mark VI (IBM-PC) $169.00
Mark VII (Auto Ans/Auto Dial) ... $119.00
Mark XII (1200 Baud) $299.00
TRS-80 Color Computer $99.00
9 Volt Power Supply $9.00

HAYES

Smartmodem 300 ..... $219.00
Smartmodem 1200 ..... $509.00
Smartmodem 1200B $459.00
Micromodem II $265.00
Micromodem II Plus $299.00
Micromodem IIE $269.00
Micromodem 100 ..... $299.00
Smart Com II $89.00
Chronograph $199.00

NOVATION

J-Cat $99.99
SmartCat 103 ..... $179.00
SmartCat 103/212 ..... $399.00
AutoCat $219.00
212 AutoCat $549.00
Apple Cat II $249.00
212 Apple Cat $569.00
Apple Cat 212 Upgrade $309.00
Cat $139.99
D-Cat $149.00
PC-Cat $339.00

ZENITH

ZT-1 ..... $309.00
ZT-10 ..... $339.00
ZT-11 ..... $369.00

APPLE INTERFACE CARDS & BUFFERS

Choose from PKASO, Orange Micro, MPC, MicroMax, Tymac, Quadram & Practical Peripherals ..... CALL

NEC PRINTERS

NEC 2050 ..... $999.00
NEC 3550 ..... $1699.00

PERCOM/TANDON DRIVES

5¼" 320K Floppy $229.00
5 Meg Hard w/Controller ... $1399.00
10 Meg Hard w/Controller ... $1699.00
15 Meg Hard w/Controller ... $2095.00
20 Meg Hard w/Controller ... $2399.00

AMDEK

310A Amber Monitor $169.00
DXY 100 Plotter $599.00
Color II $399.00

AST RESEARCH, INC.

Six Pak Plus...from $279.00
Combo Plus II...from $279.00
Mega Plus...from $309.00
I/O Plus II...from $139.00

QUADRAM

Quadlink $519.00
Quadboard...as low as $289.00
Quad 512 Plus...as low as ... $249.00
Quadcolor...as low as $219.00
Chronograph $89.00
Parallel Interface Board $89.00
64K RAM Chips Kit $59.00

MICRO PRO

WordStar/Mail Merge $349.00
InfoStar $299.00
SpellStar $159.00
CalcStar $99.00

MICROSTUF

Crosstalk $105.00

MICROSOFT

Multiplan $159.00

pfs

	APPLE	IBM
Write:	79.00	89.00
Graph:	79.00	89.00
Report:	79.00	79.00
File:	79.00	89.00
Solutions*: as low as	16.00	16.00

*Call On Titles

IBM

ASHTON-TATE

d8ASE II $389.00
Friday! $185.00

IUS

EasyWriter II $249.00
EasySpeller $119.00
EasyFiler $229.00

CONTINENTAL SOFTWARE

1st Class Mail/Form Letter ... $79.00
The Home Accountant Plus ... $88.00

SYNAPSE

File Manager $89.00

LOTUS

1-2-3 ..... $319.00

PROFESSIONAL SOFTWARE

PC Plus/The 8oss $349.00

VISICORP

	IBM	APPLE
VisiCalc		159.00
VisiCalc 4	159.00	
VisiCalc-Advanced		269.00
VisiWord/Spell	249.00	
Visitrend/Plot	199.00	199.00
VisiLink		169.00
VisiFile	199.00	169.00
VisiSchedule	199.00	199.00
Visidex		159.00
VisiPlot		135.00
VisiTerm		75.00
Desktop Plan	199.00	169.00
8us. Forecast Model	75.00	75.00
Stretch Calc	75.00	75.00
VisiTutor Calc	59.00	59.00
VisiTutor-Advanced	75.00	75.00
VisiTutor Word	59 00	59.00
Vision Calc	249.00	
Vision Graph	129.00	
Vision Mouse	159.00	
Vision Host	319.00	

PRINTERS

AXIOM

AT-100 Atari Interface $239.00
CD-100 CBM 64/VIC 20 ..... $239.00
GP-100 Parallel Interface ... $199.00

BMC

401 Letter Quality $589.00
BX-80 Dot Matrix $269.00

CENTRONICS

122 Parallel $399.00
739-1 Parallel $299.00
739-3 Serial $349.00

C.ITOH

Gorilla Banana $209.00
Prowriter B510P $379.00
Prowriter 1550P $679.00
A10 (18 cps) $569.00
F10-40 ..... $999.00
F10-55 ..... $1499.00

COMREX

ComWriter II Letter Quality ... $499.00

DIABLO

620 Letter Quality $949.00
630 Letter Quality $1749.00

DAISYWRITER

2000 ..... $999.00
Tractor Feed $109.00

EPSON

MX-80FT, MX-100, RX-80, RX-80FT, FX-80, FX-100 ..... CALL

IDS

Prism 80...For Configurations... CALL
Prism 32...For Configurations... CALL

MANNESMAN TALLY

160L $589.00
180L $799.00
Spirit 80 ..... $309.00

NEC

2010/2030 ..... $899.00
8023 Dot Matrix $379.00
8025 Dot Matrix $669.00
3510 Serial/Letter Quality ... $1449.00
3530 Parallel/Letter Quality... $1499.00
7710/7730 Serial/Parallel... $1949.00

OKIDATA

B2,B3,84,92,93,2350,2410 ... CALL

SMITH CORONA

TP-2 ..... $439.00
Tractor Feed $119.00

SILVER REED

500 Letter Quality $469.00
550 Letter Quality $699.00

STAR

Gemini 10X $299.00
Gemini P15 ..... $399.00
Delta 10 ..... $559.00
Serial Board $75.00

TOSHIBA ..... CALL
TRANSTAR ..... CALL

PRINTER CABLES

Available for Atari, Commodore, IBM, Apple, Epson, Kaypro, Televideo, Franklin, Eagle, Sanyo, Osborne, NEC, Zenith and many others. We supply all your computer needs!

PAPER SUPPLIES

1000 shts. 8½x11 Tractor Paper...19.99
1000 shts. 14½x11 Tractor Paper...$24.99
1 or 2" Address Labels $9.99

=WEST=

800-648-3311

In NV call (702)588-5654 Dept. 0215
Order Status Number: 588-5654
P.O. Box 6689, Stateline, NV 89449

=CANADA=

800-268-4559

In Toronto call (416)828-0866, Dept 0215
Order Status Number: 828-0866
2505 Dunwin Drive, Unit 1B
Mississauga, Ontario, Canada L5L1T1

=EAST=

800-233-8950

In PA call (717)327-9575, Dept 0215
Order Status Number: 327-9576
Customer Service Number: 327-1450
477 E. Third St., Williamsport, PA 17701

VISA

No risk, no deposit on C.O.D. orders. Pre-paid orders receive free shipping within the UPS Continental United States with no waiting period for certified checks or money orders. Add 3% (minimum $5.00) shipping and handling on all C.O.D. and credit card orders. Larger shipments may require additional charges. NV and PA residents add sales tax. All items subject to availability and price change. We stock manufactuer's and third party software for most all computers on the market. Call today for our new catalog.

COMPUTER MAIL ORDER

FRANKLIN

ACE 1000 Color Computer ....CALL
ACE Family Pack System .....CALL
ACE PRD PLUS System .....CALL
ACE 1200 Office Mgmt. System ...CALL
"NOT IIE EXPENSIVE"

APPLE/FRANKLIN DISK DRIVES

MICRO-SCI

A2 .....$219.00
A40 .....$299.00
A70 .....$319.00
C2 Controller $79.00
C47 Controller $89.00

RANA

Elite 1 .....$279.00
Elite 2 .....$389.00
Elite 3 .....$569.00

APPLE IIe STARTER PACK

64K Apple IIe, Disk Drive & Controller, 80 Column Card, Monitor II & DOS 3.3
COMPLETE $1199

HOME COMPUTERS

ATARI

600XL..... $199
800XL..... $299

1200XL .... CALL
1400XL .... CALL

1010 Recorder $74.00
1020 Color Printer $249.00
1025 Dot Matrix Printer.....$449.00
1027 Letter Quality $299.00
1030 DirectConnect Modem...CALL
1050 Disk Drive $379.00
CX30 Paddle $12.00
CX40 Joystick.....each...$8.00
CX77 Touch Tablet $64.00
CX80 Trak Ball $48.00
CX85 Keypad $105.00
488 Communicator II.....$229.00
4003 Assorted Education....$47.00
4011 Star Raiders $33.00
4012 Missile Command $29.00
4013 Asteroids $29.00
5049 VisiCalc $159.00
7097 Logo $79.00
7101 Entertainer $69.00
7102 Arcade Champ $75.00
8026 Dig Dug $33.00
8030 E.T. Phone Home $33.00
8031 Donkey Kong $39.00
8033 Robotron $35.00
8034 Pole Position $39.00
8036 Atari Writer $79.00
8040 Donkey Kong, Jr. $39.00
8043 Ms. PacMan $39.00
8044 Joust $39.00

DISKETTES

MAXELL

5¼" MD-1 .....$29.00
5¼" MD-2 .....$39.00
8" FD-1 (SS/DD) $39.00
8" FD-2 (DS/DD) $49.00

VERBATIM

5¼" SS/DD $26.99
5¼" DS/DD $36.99

ELEPHANT

5¼" SS/SD $18.49
5¼" SS/DD $22.99
5¼" DS/DD $28.99

HEAD

5¼" Disk Head Cleaner $14.99

DISK HOLDERS

INNOVATIVE CONCEPTS

Flip-n-File 10 .....$3.99
Flip-n-File 50 .....$17.99
Flip-n-File(400/800ROM)Holder..$17.99

LJK ENTERPRISES

Atari Letter Perfect-Disk(40/80)...$79.99
Atari Letter Perfect-ROM(40 col)...$79.99
Atari Letter Perfect-ROM(80 col)...$79.99
Atari Data Perfect-ROM(80 col).....$79.99
Atari Spell Perfect-DISK $59.99
Atari Utility/MailMerge $21.00
Apple Letter Perfect.....$99.00
Apple Data Perfect $75.00
Apple LJK Utility $21.00
Apple Lower Case Generator ...$19.00

PERCOM

AT 88-S1 .....$329.00
AT 88-A2 .....$259.00
AT 88-S2 .....$529.00
AT 88-S1PD $429.00
AT 88-DDA $119.00
RFD 40-S1 .....$449.00
RFD 40-A1 .....$269.00
RFD 40-S2 .....$699.00
RFD 44-S1 .....$539.00
RFD 44-S2 .....$869.00

TEXAS INSTRUMENTS

TX 99-S1 .....$279.00

RANA

1000.....$319.00

TRAK

AT-D2.....$389.00

MEMORY BOARDS

Axlon 32K $59.00
Axlon 48K $99.00
Axlon 128K.....$299.00
Intec 32K $59.00
Intec 48K $85.00
Intec 64K $99.00
Intec Real Time Clock $29.00

ALIEN VOICE BOX

Atari $119.00
Apple.....$149.00

KOALA PAD

Atari.....$75.00
Apple $85.00
IBM $95.00
CBM 64.....$75.00

CONTROLLERS & JOYSTICKS

WICO

Joystick.....$21.99
3-way Joystick.....$22.99
Famous Red Ball.....$23.99
Power Grip.....$21.99
BOSS Joystick $17.99
ATARI/VIC Trak Ball.....$34.99
Apple Trak Ball.....$54.99
Apple Adapter.....$15.99
Apple Analog $37.99

KRAFT

Joystick.....$41.99
Atari Single Fire.....$12.99
Atari Switch Hitter.....$15.99
Apple Paddles.....$34.99
IBM Paddles.....$34.99
IBM Joystick $46.99

AMIGA

3100 Single $13.99
3101 Pair $19.99
Joyboard $37.99

TG

Atari Trak Ball $47.99
Apple Joystick $47.99
Apple Trak Ball.....$47.99

commodore

C 64 $199

CBM 4032.....$599.00
CBM 8096 .....$869.00
CBM 9000.....$999.00
B128-80.....$769.00
CBM 64K Memory Board....$269.00
8032 to 9000 Upgrade $269.00
2031LP Disk Drive $299.00
8050 Disk Drive.....$949.00
8250 Disk Drive $1199.00
4023 Printer.....$379.00
8023 Printer.....$569.00
6400 Printer $1399.00
Z-RAM $499.00
Silicon Office $699.00
The Manager.....$199.00
Soft ROM.....$125.00
VisiCalc $159.00

PROFESSIONAL SOFTWARE

Word Pro 2 Plus.....$159.00
Word Pro 3 Plus.....$189.00
Word Pro 4 Plus/5 Plus...each...$279.00
InfoPro $179.00
Administrator $399.00
Power.....$79.00

MSD

SD1 Disk Drive $349.00

VIC 20 .....CALL
CBM 8032..... $599

C1541 Disk Drive $249.00
C1530 Datasette $69.00
C1520 Color Printer/Plotter...$169.00
M-801 Dot Matrix/Parallel...$219.00
C1526 Dot Matrix/Serial $279.00
C1702 Color Monitor $249.00
C1311 Joystick $4.99
C1312 Paddles $11.99
C1600 VIC Modem $59.00
C1650 Auto Modem.....$89.00
Logo 64.....$49.00
Pilot 64 .....$39.00
Simon's Basic $19.00
Word Pro 64 Plus.....$59.00
Parallel Printer Interface.....$49.00
Calc Result 64 .....$129.00
Codewriter 64 .....$75.00
Quick Brown Fox.....$49.00
Word Pro 64 Plus.....$59.00

We stock a full inventory of software for Commodore, such as: Artworx, Broderbund, Commercial Data, Creative Software, Epyx, HES, MicroSpec, Nufekop, Romox, Sirius, Synapse, Thorn EMI, Tronix, UMI, Victory, Spinnaker, Rainbow & Timeworks!
CALL FOR DETAILS!

HANDHELD COMPUTERS

hp

41CX $249.99
41CV $199.99
41C $144.99

HP 10C $51.99
HP 11C $69.99
HP 12C $88.99
HP 15C $88.99
HP 16C $88.99
HP 75C $749.99
HPIL Module $98.99
HPIL Cass. or Printer $359.99
Card Reader.....$143.99
Extended Function Module ...$63.99
Time Module $63.99

NEC

PC-82D1 Personal Computer...$599.00
PC-8221A Thermal Printer....$149.00
PC-8281A Data Recorder $99.00
PC-8201-06 8K RAM Chips....$105.00
PC-8206A 32K RAM Cartridge...$329.00

SHARP

PC-1500A...$165.99
PC-1250A....$88.99

CE-125 Printer/Cassette....$128.99
CE-150 Color Printer/Cassette...$171.99
CE-155 8K RAM $93.99
CE 161 16K RAM $134.99
CE 500 ROM Library $29.99

TIMEX/SINCLAIR

Timex/Sinclair 1000 .....$24.99
Timex/Sinclair 2068 .....CALL
16K Memory.....$25.00
2040 Printer.....$99.99
VuCalc $17.99
Mindware Printer.....$99.99

=WEST= 800·648·3311

In NV call (702)588-5654. Dept 0215
Order Status Number: 588-5654
P.O. Box 6689, Stateline, NV 89449

=CANADA= 800·268·4559

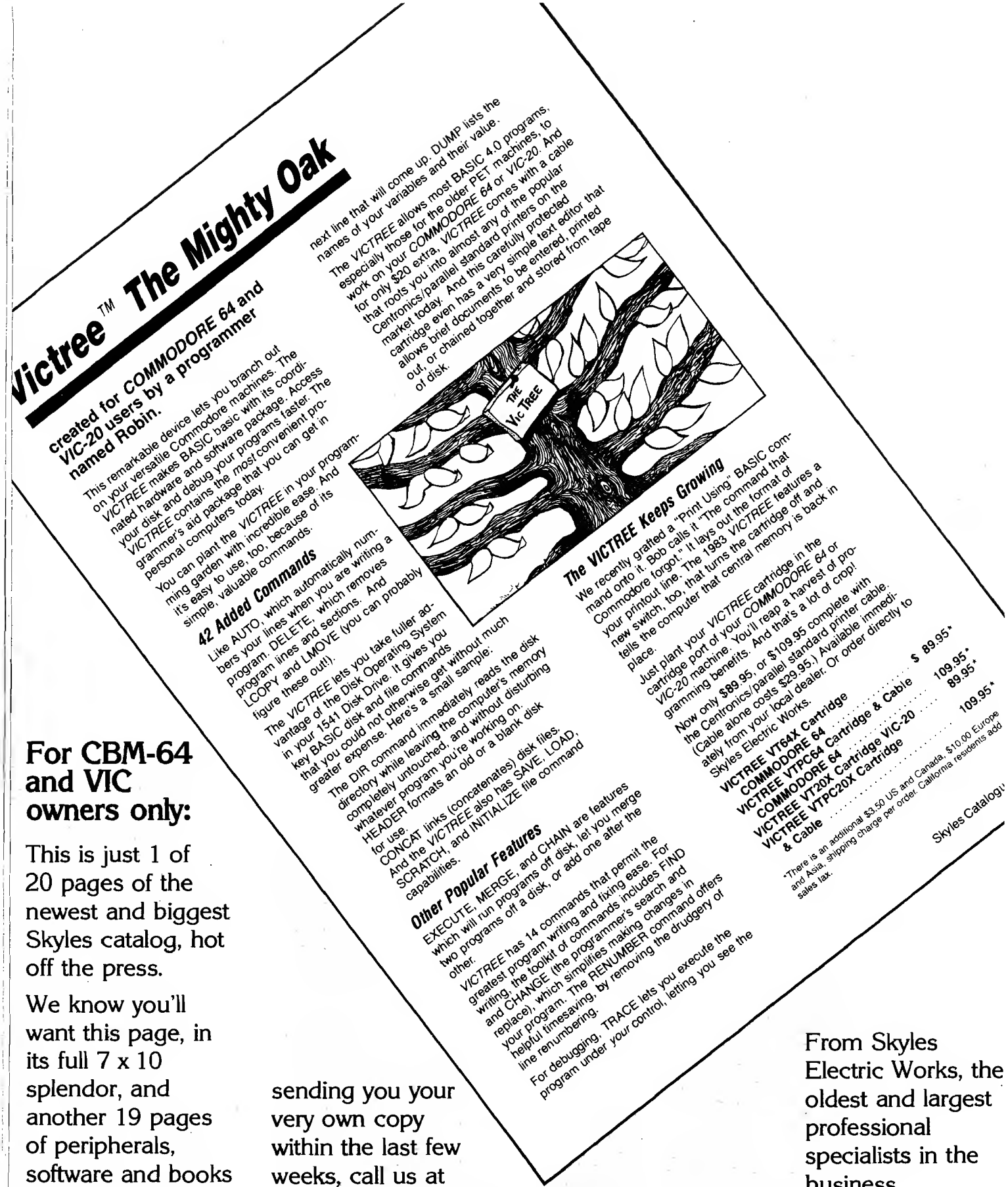
In Toronto call (416)828-0866. Dept. 0215
Order Status Number: 828-D866
2505 Dunwin Drive, Unit 1B
Mississauga, Ontario, Canada L5L1T1

=EAST= 800·233·8950

In PA call (717)327-9575 Dept 0215
Order Status Number: 327-9576
Customer Service Number: 327-1450
477 E. Third St., Williamsport, PA 17701

MasterCard

CANADIAN ORDERS: All prices are subject to shipping, tax and currency fluctuations. Call for exact pricing in Canada.
INTERNATIONAL ORDERS: All shipments outside the continental United States must be pre-paid by certified check only. Include 3% (minimum $5.00) shipping and handling. EDUCATIONAL DISCOUNTS: Additional discounts are available to qualified Educational Institutions.
APO & FPO: Add 3% (minimum $5.00) shipping and handling.



Victree™ The Mighty Oak

created for COMMODORE 64 and VIC-20 users by a programmer named Robin.

This remarkable device lets you branch out on your versatile Commodore machines. The *VICTREE* makes BASIC with its coordinated hardware and software package. Access your disk and debug your programs faster. The *VICTREE* contains the *most* convenient programmer's aid package that you can get in personal computers today.

You can plant the *VICTREE* in your programming garden with incredible ease. And it's easy to use, too, because of its simple, valuable commands.

42 Added Commands

Like AUTO, which automatically numbers your lines when you are writing a program. DELETE, which removes program lines and sections. And LCOPY and LMOVE (you can probably figure these out!).

The *VICTREE* lets you take fuller advantage of the Disk Operating System in your 1541 Disk Drive. It gives you key BASIC disk and file commands that you could not otherwise get without much greater expense. Here's a small sample:

The DIR command immediately reads the disk directory while leaving the computer's memory completely untouched, and without disturbing whatever program you're working on.

HEADER formats an old or a blank disk for use.

CONCAT links (concatenates) disk files. And the *VICTREE* also has SAVE, LOAD, SCRATCH, and INITIALIZE file command capabilities.

Other Popular Features

EXECUTE, MERGE, and CHAIN are features which will run programs off disk, let you merge two programs off a disk, or add one after the other.

VICTREE has 14 commands that permit the greatest program writing and fixing ease. For writing, the toolkit of commands includes FIND and CHANGE (the programmer's search and replace), which simplifies making changes in your program. The RENUMBER command offers helpful timesaving, by removing the drudgery of line renumbering.

For debugging, TRACE lets you execute the program under your control, letting you see the next line that will come up. DUMP lists the names of your variables and their value.

The *VICTREE* allows most BASIC 4.0 programs, especially those for the older PET machines, to work on your *COMMODORE 64* or *VIC-20*. And for only $20 extra, *VICTREE* comes with a cable that roots you into almost any of the popular Centronics/parallel standard printers on the market today. And this carefully protected cartridge even has a very simple text editor that allows brief documents to be entered, printed out, or chained together and stored from tape of disk.

The VICTREE Keeps Growing

We recently grafted a "Print Using" BASIC command onto it. Bob calls it "The command that Commodore forgot." It lays out the format of your printout line. The 1983 *VICTREE* features a new switch, too, that turns the cartridge off and tells the computer that central memory is back in place.

Just plant your *VICTREE* cartridge in the cartridge port of your *COMMODORE 64* or *VIC-20* machine. You'll reap a harvest of programming benefits. And that's a lot of crop!

Now only **$89.95**, or **$109.95** complete with the Centronics/parallel standard printer cable. (Cable alone costs $29.95.) Available immediately from your local dealer. Or order directly to Skyles Electric Works.

VICTREE VT64X Cartridge	**$ 89.95***
COMMODORE 64 Cartridge & Cable	**109.95***
VICTREE VTPC64 Cartridge	**89.95***
COMMODORE 64 Cartridge VIC-20	**109.95***
VICTREE VT20X Cartridge	
VICTREE VTPC20X Cartridge & Cable	

*There is an additional $3.50 US and Canada, $10.00 Europe and Asia, shipping charge per order. California residents add sales tax.

Skyles Catalog

For CBM-64 and VIC owners only:

This is just 1 of 20 pages of the newest and biggest Skyles catalog, hot off the press.

We know you'll want this page, in its full 7 x 10 splendor, and another 19 pages of peripherals, software and books that will make your CBM-64 or VIC computer even nicer to live with. So, if we missed sending you your very own copy within the last few weeks, call us at **(800) 227-9998 (unless you live in California, in which case call (415) 965-1735.**

From Skyles Electric Works, the oldest and largest professional specialists in the business.

Skyles Electric Works
231-E S. Whisman Road
Mountain View, California 94041

Listing 2

```
0 PRINT"{CLEAR,DOWN6}   NOTE: DOSPLUS SHOULD BE IN MEMORY"
1 PRINT"{DOWN3}            DOSPLUS+ LOADER{DOWN3}":PRINT"            WAIT
  --LOADING"
10 DIM N(100): I=0
20 READ A$: I=I+1: CS=0
30 X$=LEFT$(A$,2): GOSUB 100: IF Z<1 THEN SYS 51200: END
40 BY=Z: KS=Z: X$=MID$(A$,3,4): GOSUB 100: LC=Z
50 X$=MID$(A$,3,2): GOSUB 100: KS=KS+Z: X$=MID$(A$,5,2): GOSUB 100: KS=KS+Z
60 FOR J=1 TO BY: X$=MID$(A$,5+J*2,2): PRINT".";
70 GOSUB 100: N(J)=Z: CS=CS+Z: NEXTJ
80 X$=MID$(A$,7+BY*2,4): GOSUB 100: IF Z<>CS+KS THEN PRINT"ERROR LINE #";I:
   STOP
90 FOR J=1 TO BY: POKE LC+J-1,N(J): NEXT J:PRINT"*": GOTO 20
100 :
110 REM HEX TO DECIMAL
120 REM X$ IS HEX NUMBER TO BE CONVERTED
130 REM Z IS DECIMAL NUMBER CONVERTED
140 Z=0
150 L=LEN(X$): FORK=1TOL
160 Y=ASC(MID$(X$,K,1))
190 Z=Z*16+Y-48+7*(Y>57)
200 NEXT K: RETURN
1000 DATA"18CF59EC83CFF0228E83CF78A50129FE8501BD20A08D88CFBC00A00DF8"
1010 DATA"18CF71A9C08D8BCF2084CFA501090185015B4C00C000A200BD00A10AB5"
1020 DATA"18CF899D00C0E8D0F7EE88CFEE8BCF88D0EE60A20020A0CFA201BD1040"
1030 DATA"18CFA110C88D88CFBD18C88D8BCFA0044C84CF78A50129FD8501A20D77"
1040 DATA"18CFB90220A0CFA20320A0CFA501090285015B6078A50129FD85010A1E"
1050 DATA"18CFD1A20420A0CFA20520A0CF20E1CF1890E1A204BC10C8C8C8C80E0E"
1060 DATA"16CFE98CF3CF8CFACFA2E8BD00FB8D20D0E8BD00FB8D21D0600FAE"
1070 DATA"18C81004D8F0F4F8FCE8ECF0F404D804D804D8AD20D048AD21D0480FBB"
1080 DATA"18C8282099CF20CACF20E4FFC90DD0F920B1CF688D21D0688D20D00E56"
1090 DATA"18C84060A0068CD2CF8CE2CFC88CD7CF2020C8A0048CD2CF8CE2CF0FA0"
1100 DATA"05C858C88CD7CF60047F"
1110 DATA"18CB602099CFA90085FD85FE2041CB0D4E554D424552203F2000200A1A"
1120 DATA"18CB78CFFFC90DF03CC924D0034C60C8C93090DAC93AB0D6290FA20E25"
1130 DATA"18CB9011D005900269094A66FE66FDCAD0F420CFFFC90DD0DF20410DD0"
1140 DATA"15CBA8CB20202400A5FE20CBC8A5FD20CBC84C63CB4CB1CF0CA8"
1150 DATA"18C85D4C63CB20CFFFC90DF022C93090F2C93A900AC94190EAC9470E39"
1160 DATA"18C875B0E6E9360A0A0A0AA2040A26FD26FECAD0F8F0D72041CB200CCE"
1170 DATA"18C88D2000A007A23038A5FDF9C2C84888A5FEF9C4C8900985FE680EDF"
1180 DATA"18C8A585FDE8C8D0E8688A8CFF0220D2FFACFF028810D8A5FD09300FD7"
1190 DATA"18C8BD20D2FF4C63CB0A006400E803102748A4A4A4A20D6C868290A57"
1200 DATA"18C8D50FC90AB0040930D00269364CD2FF2041CB20444F53504C550A35"
1210 DATA"13C8ED5320432E31393834204D4B455259414E0D00600626"
1220 DATA"01CBC82001B4"
1230 DATA"01CBD54101E2"
1240 DATA"01CBCE6001FA"
1250 DATA"01CBDEE3028D"
1260 DATA"01CBE8C8027C"
1270 DATA"01CBF5C80289"
1280 DATA"01CBEECB0285"
1290 DATA"01CBFEC80292"
1300 DATA"00"
1500 :
1510 REM THE ORIGINAL DOSPLUS SHOULD BE RESIDENT IN MEMORY
     PRIOR TO RUNNING THIS
```

(article continues on page 49)

Subscribe to MICRO... Save 20% and we'll send you a BONUS GIFT with your subscription!

Fill out the attached card and mail today!

CHOOSE FROM:

★ **Best of MICRO Vols. 2 and 3**
An $18.00 Value — FREE!

or

★ **MICRO Calc** for Apple, Atari, TRS-80, or C64
A $14.95 Value — FREE!

or

★ **MICRO on the APPLE, Vol. 3**
A $24.95 Value for $12.00

TWO NEW FROM

Don't Miss These Great MICRO Publications

A Feast Of Computing Ideas

Use These Postage-Paid Cards To Order

SAVE 20%!

Use This Postage-Paid Card To Subscribe...

Or Call Toll-Free

1-800-345-8112

(In PA 1-800-662-2444)

(MasterCard and VISA Accepted)

MICRO™ SAVE 20%

☐ YES! Enter my subscription to MICRO for just $24.00! and with it send me the BONUS GIFT I've checked.

☐ **Best of MICRO Volumes 2 *and* 3.** An $18.00 Value — **FREE!**

☐ **MICRO Calc** for ☐ Apple ☐ Atari ☐ TRS-80 ☐ C64
A $14.95 Value — **FREE!**

☐ **MICRO on the APPLE — Vol. 3** A $24.95 value — **Only $12.00**

Please rush my subscription and the BONUS GIFT I've checked to:

Name _____

Address _____

City _____ State _____ Zip _____

(Allow 6-8 weeks for delivery)

MICRO™'s Newest Books

BOOKS MICRO!

NEW for VIC-20 Users!

Mastering Your VIC-20

With 8 BASIC Projects

A book that makes learning to program your VIC-20 fun! Contains 8 projects and programs. Games, utilities — even a VIC-20 version of "VisiCalc." All 8 programs on cassette to help you learn faster.

☐ Mastering the VIC-20 @ $19.95

NEW for OSI Users!

MICRO on the OSI

Includes Machine-Language enhancements and BASIC Aids, hardware modifications for enhanced/reversed video, programs for control code and upper/lower case entry. A valuable programming tool.

☐ MICRO on the OSI — Now Only $14.95

Please rush the MICRO Books I've checked above to:

Name _____

Address _____

City _____ State _____ Zip _____

(Allow 6-8 weeks for delivery)

MA residents add 5% sales tax.

I'm paying by: ☐ Check ☐ MO ☐ VISA ☐ MC

Total Enclosed: $ _____
(Add $2.00 s/h per book.)

Visa/MC # _____

Exp. Date: _____

MICRO™'s Best Sellers

The Best of MICRO

Make your computer reference library complete by adding the "Best of MICRO" to your present book collection. Each volume full of articles and programs pulled from the pages of MICRO. Excellent software at a very low price!

☐ BOM Volume 2 @ $8.00
☐ BOM Volume 3 @ $10.00

What's Where in the Apple

Revised new addition with text added to the original atlas and gazetter. All Apple users will find this book helpful in understanding their machine and essential for mastering it.

☐ What's Where in the Apple @ $24.95
☐ The Guide @ $9.95

Best Sellers for APPLE Users!

MICRO on the APPLE

Programming aids, utilities, games, enhancements. Together Volumes 1, 2, and 3 contain over 100 programs on diskette. Fully documented and explained.

☐ 3-Volume Gift-Boxed @ $59.95
☐ Vol.1 ☐ Vol.2 ☐ Vol.3 $24.95 ea.

Please rush the MICRO Books I've checked above to:

Name _____

Address _____

City _____ State _____ Zip _____

(Allow 6-8 weeks for delivery)

MA residents add 5% sales tax.

I'm paying by: ☐ Check ☐ MO ☐ VISA ☐ MC

Total Enclosed: $ _____
(Add $2.00 s/h per book.)

Visa/MC # _____

Exp. Date: _____

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 60, CHELMSFORD, MA

POSTAGE WILL BE PAID BY ADDRESSEE

MICRO™
P.O. Box 6502
Chelmsford, MA 01824

SAVE 20%!!!!

Subscribe to MICRO

Use This Postage Paid Card to Order the Next 12 Issues of MICRO and SAVI $6.00 Off Newsstand Price!

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 60, CHELMSFORD, MA

POSTAGE WILL BE PAID BY ADDRESSEE

MICRO™
P.O. Box 6502
Chelmsford, MA 01824

A Feast O Computin Ideas...

New Book From MICRO

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 60, CHELMSFORD, MA

POSTAGE WILL BE PAID BY ADDRESSEE

MICRO™
P.O. Box 6502
Chelmsford, MA 01824

Order These Best Sellers From MICRO

Listing 3

```
10 REM*********************************
20 REM*                               *
30 REM* DOSPLUS+ LOADER   M.J.KERYAN  *
40 REM*                   12-06-83    *
50 REM* SETS UP HELP AND USRHELP      *
60 REM* SCREENS, LOADS ML ROUTINES    *
70 REM* INCLUDING DOS 5.1 WEDGE       *
80 REM*  FOR HELP PRESS RESTORE, H    *
90 REM*********************************
1000 IF A=0 THEN A=1: LOAD " DOSPLUS+.ML",8,1
1010 PRINT"{HOME}RUN 1500:"
1020 POKE 198,2
1030 POKE 631,19: POKE 632,13
1040 SYS 51200
1400 REM*****************************
1410 REM*                           *
1420 REM* USRHELP SCREEN  CALL BY    *
1430 REM*   RESTORE, U  OR SYS 51265 *
1440 REM*                           *
1450 REM*****************************
1500 POKE 53280,0: POKE 53281,0
1510 PRINT"{GREEN,CLEAR,DOWN11}        {RVS}
     PLACE YOUR SCREEN HERE{RVSOFF}"
1520 POKE 53128,4: POKE 53131,232
1530 SYS 53164
1540 POKE 60392,PEEK(53280)
1550 POKE 60393,PEEK(53281)
1560 POKE 53128,216: POKE 53131,236
1570 SYS 53164
1900 REM*****************************
1910 REM*                           *
1920 REM*  HELP SCREEN    CALL BY    *
1930 REM*   RESTORE, H  OR SYS 51232 *
1940 REM*                           *
1950 REM*****************************
2000 POKE 53280,5: POKE 53281,1
2010 PRINT"{CLEAR}              {RED,RVS}
     COMMAND SUMMARY{RVSOFF}"
2020 PRINT"    DOS 5.1              DOSPLUS 1.1"
2030 PRINT" ---------------   -------------------"
2040 PRINT"{BLACK} {UP ARROW}PGM{BLUE} LOAD & RUN    {BLACK}
     RESTORE{BLUE} STOP SCROLL"
2050 PRINT"{BLACK} /PGM{BLUE} LOAD PGM"
2060 PRINT"{BLACK} %PGM{BLUE} LOAD ML PGM
               --FOLLOW BY:"
2070 PRINT
2080 PRINT"{BLACK} {BACK ARROW}PGM{BLUE} SAVE PGM      {BLACK}A
     {BLUE} APPEND PGMS"
2090 PRINT
2100 PRINT"{BLACK} @$(BLUE}   LIST DIR      {BLACK}B
     {BLUE}  BACKGND COLOR"
2110 PRINT"                    {BLACK}C
     {BLUE}  CHARACTER COLORS"
2120 PRINT"{BLACK} @NO:NAME,ID        E
     {BLUE}  EDGE COLORS"
2130 PRINT"      FORMAT DISK"
2140 PRINT"{BLACK} @RO:NEWNM=OLDNM    D
     {BLUE}  DUMP TO PRINTER"
2150 PRINT"      RENAME DISK   {BLACK}O
     {BLUE}  PRINTER OFF"
2160 PRINT"{BLACK} @CO:NEWPGM=OLDPGM  P
     {BLUE}  PRINTER ON"
```

```
2170 PRINT"       COPY PGM
2180 PRINT"{BLACK} @S0:PGM            H
     {BLUE}  HELP (THIS LIST)"
2190 PRINT"       SCRATCH PGM   {BLACK}U
     {BLUE}  USER HELP"
2200 PRINT"{BLACK} @I{BLUE}   INITIALIZE"
2210 PRINT"{BLACK} @UI{BLUE}  RESET         {BLACK}N
     {BLUE}  NUMBER CONVERSN."
2220 PRINT"{BLACK} @V{BLUE}   VALIDATE"
2230 PRINT"{BLACK} @Q{BLUE}   QUIT DOS 5.1  {BLACK}RETURN
     {BLUE}  CANCEL"
2240 PRINT
2250 PRINT"       {RED}PRESS {RVS}RETURN
     {RVSOFF} TO CONTINUE";
3000 POKE 53128,4: POKE 53131,248
3010 SYS 53164
3020 POKE 64488,PEEK(53280)
3030 POKE 64489,PEEK(53281)
3040 POKE 53128,216: POKE 53131,252
3050 SYS 53164
3060 FOR I= 1 TO 5000: NEXT
3070 POKE 53280,6: POKE 53281,12: PRINT"{BLACK,CLEAR}"
3080 PRINT"{CLEAR}PRINT (FRE(0)+65535){DOWN,LEFT12}BYTES FREE"
3090 POKE 198,2
3100 POKE 631,19: POKE 632,13: POKE 633,17: POKE 634,17
3200 NEW
```

MICRO™

Master Directory for the Apple
Part 2

by Charles Hill

Keep all your disk directories on a master file: Sort and Print the file for a quick reference to all your programs.

Editor's Note: In issue 67 [December], page 47, we published the main article of this title. Since the listing was much longer than usual, we printed the main routines that month and promised to include the balance in the next issue. Unfortunately, we were unable to print the promised print and sort routines until this issue. We think you will find them very useful.

```
*     SORT ROUTINE
*
SORTCAT  LDA #$34          ;INV "4"
         STA $6B2
         JSR HOME
         LDX #SORTMES
         LDY /SORTMES
         JSR PRINTMES
^2       JSR GETLNZ        ;GET THE SORT FIELDS
         TXA
         BEQ )3
         LDY #$00
^1       LDA INBUFF,Y      ;MOVE THE SORT FIELDS
         CMP #$8D
         BEQ SORT
         AND #$03          ;STRIP SORT FIELDS TO
                            A RANGE OF 1 - 3
         STA FLDPOS,Y
         INY
         CPY #$03
         BNE (1
SORT     LDA #00           ;PUT A FLAG AT THE END
         STA FLDPOS,Y
         LDA NUMBER        ; MAX:= NUMBER
         STA MAX
         LDA NUMBER+1
         STA MAX+1
NXTMAX   LSR MAX+1         ;MAX:= MAX /2
         ROR MAX
         LDA MAX           ;TEST FOR MAX = 0
         ORA MAX+1
         BNE )1
^3       LDA #$B4          ;NORM "4"
         STA $6B2
         JMP COMMAND       ;END IF MAX = 0
```

```
*1      LDA #$01        ;J:= 1
        STA J
        LDA #$0
        STA J+1
        SEC             ;K:= NUMBER - MAX
        LDA NUMBER
        SBC MAX
        STA K
        LDA NUMBER+1
        SBC MAX+1
        STA K+1
TESTJK  LDA K           ;COMPARE K TO J
        CMP J
        LDA K+1
        SBC J+1
        BLT NXTMAX
        LDA J           ;H:= J
        STA H
        LDA J+1
        STA H+1
NEXTV   CLC             ;V:= H + MAX
        LDA H
        ADC MAX
        STA V
        LDA H+1
        ADC MAX+1
        STA V+1
        LDY H+1         ;FIND LOCATION OF HTH
                         ELEMENT
        LDX H
        CPX #0          ;REDUCE H TO A RANGE OF
        BNE )1          ; 0 TO NUMBER -1
        DEY
*1      DEX
        JSR MULT
        CLC
        TXA
        ADC #NAMEBUFR
        STA HPTR
        TYA
        ADC /NAMEBUFR
        STA HPTR+1
        LDY V+1         ;FIND LOCATION OF VTH
                         ELEMENT
        LDX V
        CPX #0          ;REDUCE V TO A RANGE OF
        BNE )1          ;0 TO NUMBER - 1
        DEY
*1      DEX
        JSR MULT
        CLC
        TXA
        ADC #NAMEBUFR
        STA VPTR
        TYA
        ADC /NAMEBUFR
        STA VPTR+1
        LDA #$FF        ;INITIALIZE SORT FIELD
                         POINTER
        STA FLDPTR
NXTFLD  INC FLDPTR
        LDY FLDPTR
        CPY #$03        ;MAXIMUM OF THREE
                         FIELDS
        BEQ NEXTJ       ;TEST FOR END OF FIELDS
        LDA FLDPOS,Y
        BEQ NEXTJ       ;TEST FOR END OF FIELDS
        TAY
        DEY
        LDA FLDLEN,Y    ;GET THE FIELD LENGTH
        STA MAXMY
        TYA             ;SET ZERO/NOTZERO FLAG
        BEQ CMPRID      ;CHECK FOR SPECIAL CASE
                         OF DISK ID COMPARE
*1      LDA (HPTR),Y    ;DO NORMAL COMPARE
        CMP (VPTR),Y
        BEQ )2
        BCC NEXTJ       ;DON'T SWAP THEM
```

```
        BCS SWAP
^2      INY
        CPY MAXMY
        BNE (1
        BEQ NXTFLD          ;THEY WERE EQUAL
CMPRID  LDA (HPTR),Y        ;COMPARE TWO DISK
                             ID'S (INDIRECT)
        JSR POINTID         ;POINT TO FIRST ID
        LDA IDBUFFR
        STA BUFFER
        LDA IDBUFFR+1
        STA BUFFER+1
        LDA (VPTR),Y
        JSR POINTID         ;POINT TO THE OTHER
^2      LDA (BUFFER),Y      ;COMPARE THEM
        CMP (IDBUFFR),Y
        BEQ )1
        BCC NEXTJ           ;NO SWAP NEEDED
        BCS SWAP
^1      INY
        CPY MAXMY
        BNE (2
        BEQ NXTFLD          ;THEY WERE EQUAL
SWAP    LDY #$0             ;SWAP THE RECORDS
^1      LDA (HPTR),Y
        PHA
        LDA (VPTR),Y
        STA (HPTR),Y
        PLA
        STA (VPTR),Y
        INY
        CPY #$20
        BNE (1
        SEC                 ;H:= H - MAX
        LDA H
        SBC MAX
        STA H
        LDA H+1
        SBC MAX+1
        STA H+1
        BCC NEXTJ           ;IF H ( 1
        LDA H
        ORA H+1
        BEQ NEXTJ
        JMP NEXTV
NEXTJ   INC J               ;J:= J + 1
        BNE )1
        INC J+1
^1      JMP TESTJK          ;ALWAYS
*
*
* THIS IS WHERE THE MULT ROUTINE
* FROM LAST MONTH FITS IN
*
*
SORTMES ASC "ENTER FIELDS TO SORT ON IN DECREASING
         ORDER OF IMPORTANCE  EG 312(CR)"
        HEX 8D8D
        ASC "SORT FIELDS:"
        HEX 8D
        ASC "1= DISK ID 2= FILE TYPE 3= FILE NAME"
        HEX 8D00
*
* ROUTINE TO PRINT CATALOG
*
PRINTCAT LDA #$35           ;INV "5"
        STA $702
        JSR HOME
        LDX #FLDMES         ;PRINT FIELD ORDER
                             MESSAGE
        LDY /FLDMES
        JSR PRINTMES
        JSR GETLNZ          ;GET FIELD NUMBRES
        TXA                 ;TEST FOR NULL INPUT
        BNE )2
        LDA #$B5
        STA $702
        JMP COMMAND
^2      LDX #0              ;MOVE FIELD NUMBERS
^1      LDA INBUFF,X
        CMP #$8D            ;TEST FOR END
        BEQ )2
        AND #$03            ;SAVE BITS 0 AND 1
        STA PRINTFLD,X
        DEC PRINTFLD,X      ;CONVERT RANGE TO 0 - 2
        INX
        CPX #$03            ;ANYMORE ?
        BNE (1
^2      LDA #$FF            ;SET END OF PRINT
                             FIELDS FLAG
        STA PRINTFLD,X
*
* MAIN PRINT ROUTINE
*
        LDA NUMBER          ;SET NUMBER TO PRINT
        STA COUNT
        LDA NUMBER+1
        STA COUNT+1
        LDA #NAMEBUFR       ;INIT BUFFER POINTERS
        STA BUFFER
        LDA /NAMEBUFR
        STA BUFFER+1
        LDA #$01            ;TURN ON PRINTER (PR#1)
        JSR SETOUT
PAGE    LDA #$02            ;SET NUMBER OF LINES
                             PRINTED
        STA NUMLINES        ;3 LINE TITLE
        LDY #$00            ;PRINT TITLES
^2      LDX PRINTFLD,Y      ;WHICH TITLE IS THIS ?
        BMI )1              ;IF NO MORE FIELDS
        TYA                 ;REMEMBER POSITION
        PHA
        LDY TITLEH,X        ;GET ADDRESS OF TITLE
        LDA TITLEL,X
        TAX
        JSR PRINTMES        ;AND PRINT IT
        LDA #" "
        JSR COUT            ;AND A SPACE
        PLA                 ;GET SAVED POSITION
        TAY
        INY
        CPY #$03            ;NEXT TITLE
        BNE (2
^1      JSR CROUT
NEXTLINE JSR CROUT
        LDA #$FF            ;INIT FIELD POINTER
        STA FLDPTR
NEXTFLD INC FLDPTR
        LDY FLDPTR
        CPY #$03            ;ARE WE AT THE END?
        BEQ TESTEND         ;YES - SO DO OTHER
                             CHECKING
        LDA FLDPOS,Y
        BMI TESTEND         ;OR IF THERE ARE LESS
                             THAN 3 FIELDS
        TAY
        LDA FLDLEN,Y        ;GET THE LENGTH OF
                             THE FIELD TO PRINT
        STA MAXMY
        CPY #$0C
        BEQ PRINTID         ;ARE WE PRINTING
                             A DISK ID
        CPY #$01
        BEQ PRNTTYPE        ;OR IS IT A FILE TYPE
^1      LDA (BUFFER),Y      ;PRINT FILE NAME HERE
        JSR COUT
        INY
        CPY MAXMY
        BNE (1
        LDA #" "
        JSR COUT
        BNE NEXTFLD         ;CHECK FOR THE NEXT
PRINTID LDA (BUFFER),Y      ;PRINT DISK ID HERE
        JSR POINTID         ;POINT TO THE ID
^1      LDA (IDBUFFR),Y
        JSR COUT
```

```
        INY
        CPY MAXMY
        BNE <1
        LDA #" "
        JSR COUT
        BNE NEXTFLD
PRNTTYPE LDA #" "           ;PRINT TYPE HERE
        JSR COUT            ;THIS SECTION OF CODE
                             IS "BORROWED" FROM DOS
        LDX #" "            ;PRINT AN '*' IF THE
                             FILE IS LOCKED
        LDA (BUFFER),Y
        BPL >1
        LDX #"*"
^1      TXA
        JSR COUT
        LDA (BUFFER),Y      ;PRINT THE FILE TYPE
        AND #$7F
        LDY #7
        ASL
^2      ASL
        BCS >1
        DEY
        BNE <2
^1      LDA FILETYPE,Y
        JSR COUT
        LDA #" "
        JSR COUT
        JSR COUT
        BNE NEXTFLD
TESTEND LDA COUNT           ; TEST FOR END OF NAMES
                             AND PAGE
        BNE >1
        DEC COUNT+1         ;COUNT= COUNT - 1
^1      DEC COUNT
        LDA COUNT
        AND COUNT+1
        CMP #$FF
        BNE >1              ;THERE ARE MORE
                             SO CONTINUE
        JSR SETOUTO         ;TURN OFF PRINTER
        LDA #$B5            ;NORM "5"
        STA $702
        JMP COMMAND
^1      INC NUMLINES
        CLC                 ;INCREMENT BUFFER
                             POINTER
        LDA BUFFER          ;POINT TO NEXT ENTRY
        ADC #$20
        STA BUFFER
        BCC >1
        INC BUFFER+1
^1      LDA NUMLINES        ;HAVE WE PRINTED
                             A PAGEFUL YET ?
        CMP #MAXLINE
        BCS >2
        JMP NEXTLINE        ;NO SO CONTINUE
^2      LDA #$0C            ;YES SO -)
        JSR COUT            ;FORM FEED
        JMP PAGE
FILETYPE ASC "TIABSRAL"
FLDMES  ASC "ENTER THE FIELDS TO PRINT  EG 213<CR>"
        HEX 8D
        ASC "FIELD NUMBERS:"
        HEX 8D
        ASC "1= DISK ID 2= FILE TYPE 3= FILE NAME"
        HEX 8D00
IDTITLE ASC " DISK ID"
        HEX 00
TYPTITLE ASC "TYPE"
        HEX 00
NAMTITLE ASC "          FILE NAME          "
        HEX 00
TITLEL  BYT IDTITLE,TYPTITLE,NAMTITLE
TITLEH  HBY IDTITLE,TYPTITLE,NAMTITLE
*
```

MICRO™

MICRO™

CoCo Bits

by John Steiner

Assembly Language Programming

When the winter wind blows, its a good time to pull a chair near the fire and pick up a good book. If you own a CoCo, that book might be something relating to your computer. Radio Shack has finally released that long-awaited assembly language book for the Color Computer. If you haven't seen it yet, stop by the store and look for it. Its a winner. William Barden, Jr. takes the novice through nearly every phase of assembly programming on the CoCo. The book, entitled "TRS-80 Color Computer Assembly Language Programming", is written with the beginner in mind, and uses the Edtasm+ ROM version as the assembler in programming examples. About the only negative thing I could say about the book is its dependence upon Edtasm+. If you have another assembler, you will have to translate assembler commands to the format your assembler uses. Edtasm is an excellent editor for cassette users, though, and novices might be willing to purchase it anyway. The price of the book is right, though, at $6.95 a copy. A disk version of Edtasm is due out soon.

While on the subject of Edtasm, Rainbow magazine carried an article by Roger Schrag that had patches to Edtasm that allowed it to run on disk. This patch is fairly easy to implement, and seems to work well. If you are interested, the article is in September, 1983 Rainbow. I haven't found any problems with the patch yet. If you run into any bugs while working with it, please let me know.

Color Computer Shows

The November Rainbowfest in Dallas and the Color Computer Expo in California are history as I write this. I had hoped to make it to Dallas, but conflicting schedules prevented it. Bob Rosen of Spectrum Projects gave me a first hand report on the Dallas show. There was not quite as large a crowd at this show as there was at the Chicago 'fest. This might be expected as the population is not as dense. One of the new items on display was a CP/M board that plugs into the CoCo ROM port. The board only costs $249.00. and opens up a lot of powerful business software. It will be interesting how easy it is to modify the software for use on the CoCo. An interesting piece of software was GRAPHIC.COM (sold by Spectrum Projects). Bob said it was the hit of the show (he might be prejudiced, since he sells it). He promised me a review copy, so I will pass along my thoughts on the program as well. The real hit of the show was the tour of Tandy factories where the CoCo is manufactured. I have been told by several attendees that they probably enjoyed that best.

The Color Computer Expo in Pasadena, California was crowded, according to Sue Searby of Computerware in Encinitas, California (Who says computer shows are a thing of the past?) The next big Color Computer show I know of is a Rainbowfest in Cherry Hill, New Jersey in April, 1984. If you have word on any others, please let me know. I will pass the information along.

The DRAGON Arrives

The Dragon Computer is being sold in the US by Tano Corporation. As I write this, there are a few Dragon dealers offering the machines for sale. Reports of the new machine are favorable, for the most part. Advantages include a parallel printer port, a video monitor jack, and a typewriter keyboard. Tano is providing a line of software included in price of the computer. Its biggest disadvantage is in compatibility with the CoCo. Though the Extended Color BASIC language is nearly identical in syntax, the tokens used to represent BASIC keywords are not the same, so BASIC programs written on the CoCo make little sense when loaded into the Dragon. Programs may be typed in, or transferred via modem, etc. In addition, I have received a program for review from Frank Philbrow of Elkan Electronics in Prestwich, Manchester, England. This program, called Dragon Cruncher translates BASIC programs from CoCo to Dragon and Dragon to CoCo. I am looking forward to reviewing it, however I don't have a Dragon handy. I am arranging to look one over. I will let you know more about both the computer and the program when I have had a chance to look them over.

Two More Word Processors

One of the best applications for a home computer is word processing. In the August issue, which was dedicated to word processing, I mentioned some of the word processors I had worked with. A short time later, I heard from Tom Nelson of Nelson Software Systems (now called Softlaw Corporation). His program, Super Color Writer, was one of the other major word processors in use when that was written, and Tom wanted me to look it over. The program, called VIP writer now, has been lowered in price, and costs only $59.95 in either ROM or disk versions. The software is very powerful, and is command driven, unlike Telewriter, which is menu driven. It takes a little extra effort to learn to use VIP writer, but it is well worth the effort. The package is first rate. VIP Writer supports 64K and has a bank switching technique that allows disk users 41K of workspace, and ROM users 52. If you forego the high resolution graphics screens, ROM users can use over 62K of workspace. Screen displays include the standard 32 character mode, as well as 51, 64 and 85 by 21 or 24 lines per screen. On a monitor, displays are very readable, and background color is easily changed. The editor is easy to learn and use with the well written 110 page manual. Scrolling is allowed both vertically and horizontally, with column widths up to 240. Just about any special print function can be done from the program. A format window allows you to see just exactly what the text will look like including margins, page breaks and justification. This is what I miss most in Telewriter. About the only bad things I can say about VIP Writer are the fact that it comes on a protected disk (no backups allowed) and it is a lot to learn (hardly a

disadvantage when you consider its power). It consistantly turned out good looking text.

It takes a lot to learn to use any word processor, and I am using the latest CoCo word processor to write this column. Elite Word, from Elite Software, is relatively inexpensive, simple and yet powerful in its capacity. Like VIP Writer, Elite Word can typeover text or insert text. Both programs have key repeat, which is a real advantage. Elite Word is easy to learn. I was writing complicated files, and saving and printing them in only a few hours. The editor works well, and supports the 51 by 20 line mode of text entry. This graphic screen has a smooth scroll that is easy to get used to watching. Text can be entered anywhere without long pauses while working at the beginning of a large text file. Though it takes awhile for text to appear on the screen in this situation, any keys hit are stored in a buffer, and displayed as the computer gets to them. A window displays text as it can be printed, and command help is easy because of a status line at the top of the screen. Loading and saving files is easy, and saves can be done in binary, formatted or non-formatted ASCII. The print routine seems to work perfectly, and I noted only one manual error. Default for page numbering is 1, not 0. Print routine setup is easy, and supports any brand printer because you modify a BASIC loader program with your own printers commands. One of the first things to do to implement the printer routines is edit several lines, putting the proper data from your printer manual into the loader program. The 50 page manual is sufficient, and easily readable. A variable text merge allows a person to create form letters that are printed from a data file list. If you have ever worked with Wordstar's MailMerge, you will know how useful that function can be. Disks are not protected, so you may make backups to your heart's content. I only noticed a couple of disadvantages with this program. With only one graphic screen available for editing you never see the text as it looks on the page until you view it from the format window, or print it. Most of the time, this should be adequate. You must always remember to put a carriage return at the end of the last line of text or it won't be printed. All in all, both of these programs will do a good job for you. My vote for easiest is Elite Word, and most powerful is VIP Writer.

MICRO™

Interface Clinic

by Ralph Tenny

Last month we started talking about A/D (analog to digital) converters. Let's review briefly: A/D converters measure analog (naturally occurring) voltages in discrete steps. An A/D converter which has coarse steps (low resolution) most likely will not be able to measure a voltage accurately even if it is a perfectly accurate converter. For example: measure the length of a pencil, using only the inch marks on the ruler, and express the length in terms of the nearest mark. Next, measure the same pencil using a ruler which has 1/32" divisions, again expressing the length in terms of the nearest mark. No matter how accurate the first ruler is, you can make a better measurement the second time! So, when it comes to choosing an A/D converter, the first rule is to be sure to get a unit with sufficient resolution. *(Ed. Note: Figure 3 in last month's column inadvertantly got referenced in the text as Figure 4. There was no Figure 4.)*

Suppose you want to measure voltages between 0 and 5 volts to an accuracy of + or - 1%. 1% is equal to one part in 100. If a converter has 7 bits of resolution and perfect accuracy, it would suffice, since 1/128 = .78%. A typical accuracy for a 7 bit converter is + or - 1/2 bit, which is .39%. Since .78 + .39 = 1.17%, it is not accurate enough. Let's try an 8-bit counter (1/256 or .39%) with 1 bit accuracy: .39% + .39% = .78% worst case, which will be suitable. Actually, such a converter is very inexpensive unless you want to make is high speed (speed costs)!

Figure 1 shows a very simple type of A/D converter which can be driven by a PIA. It consists of a resistor network called an R-2R ladder (the resistors used have a ratio of 2:1), followed by a voltage comparator. The resolution of this comparator will obviously be 7 bits or one part in 128. The resistive network generates an output voltage (at point 'A') proportional to the binary number output from the PIA. If you stop to think about that statement, you must realize that the PIA plus the R-2R ladder make up a simple D/A converter!

Let's get an idea how the output voltage can be directly related to the binary output of the PIA. If all the outputs are at logic zero, obviously the voltage at A will be zero. Also, it is easy to see that if all outputs are at logic one, the output will be maximum. If the resistors are very high value, this maximum output voltage will be essentially equal to the supply voltage of the PIA.

Some more examples, not immediately obvious: if Bit 6 of the PIA is high and all the rest are zero, the output is one-half of full scale or one-half of the supply voltage. If you output binary 00100000 (Bit 5 high), the output is 1/4 of full scale, etc. We have to observe one condition regarding the output of this D/A - it will not be accurate unless it feeds a very high impedance load.

Well, if the PIA and resistor ladder make a D/A converter, how does simply adding a voltage comparator make Figure 1 an A/D converter? Actually, it takes intelligent control of the PIA - software - to complete the job. Here's how it works: the unknown voltage is fed to the second comparator input. If the unknown voltage is higher that the D/A output, the comparator output will be high. Suppose you set the PIA output to binary 00000000. That will give minimum output from the D/A. If you have an input voltage lower than 5 volts and higher than a single-bit output from the D/A, the comparator output will be low. You can now increase the output of the D/A one binary count at a time until the comparator switches high. If changing only the LSB (least significant bit) of the converter up and down switches the comparator, leave that bit off and record the output binary word.

Suppose that the final binary count is 00111000. The resolution of the converter is .0391 volts (5 volts/128). 00111000 binary equals 56 decimal, so the unknown voltage is more than 2.19

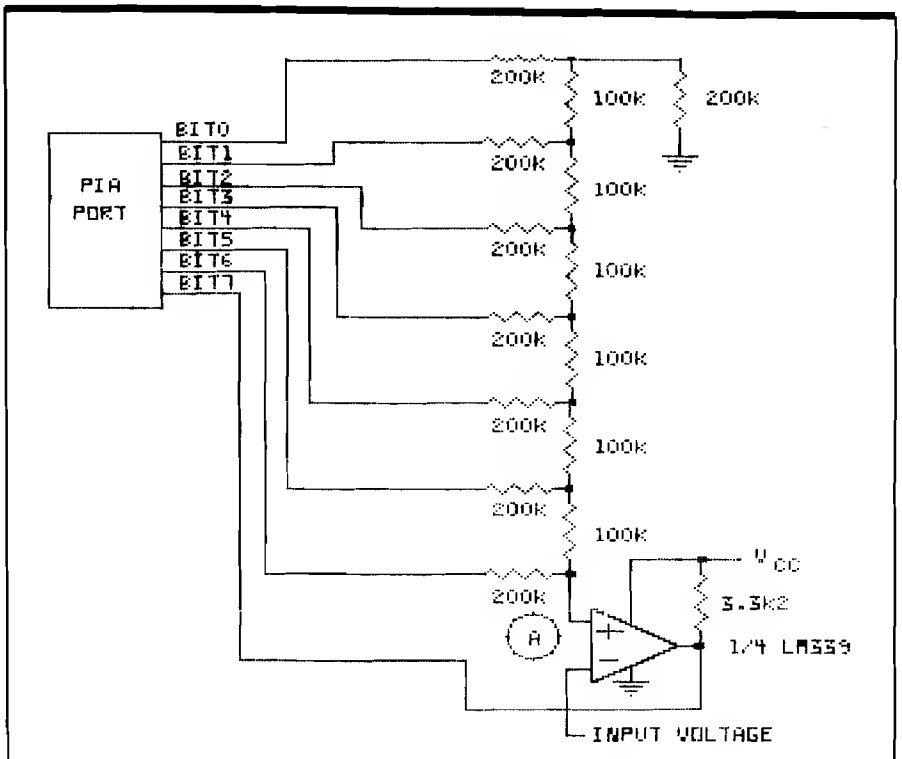


Figure 1. Several picked resistors driven by a parallel port make a D/A converter; add a comparator and software to get an A/D converter.

volts (56 * .0391), and less than 2.23 volts (57 * .0391). We can't read it any closer because of lack of resolution.

Let's get more familiar with this type of A/D converter. Build this simple circuit and measure the D/A output for different binary values. A typical digital voltmeter will give the best results, and enough resolution to make meaningful comparisons.

If your computer has a parallel output port which is a PIA (the Commodore 64 User Port, for example), try building the R-2R ladder with the resistor ends soldered to a connector which fits the output port. Make some preliminary measurements to find out the full scale output of the converter. You will probably find that the output isn't exactly 5 volts. It may be slightly lower or higher, depending upon the Vcc voltage fo the computer. You can't easily correct this error, except to compute the error and allow for it if you need better accuracy.

If your computer doesn't have a parallel port, use the Serial-Parallel adapter we experimented with in this column recently (November **MICRO 66:106**). If your computer's parallel port uses TTL parts, the accuracy will be considerably degraded since TTL parts do not pull up to Vcc or down to 0 volts. Figure 2 shows one possible fix. Use a CMOS buffer to drive the R-2R ladder. The output levels will now be Vcc and 0 volts. Also, if you wish to adjust the full scale accuracy, carefully adjust the Vcc for the CMOS buffer so the full scale output is exactly 5 volts.

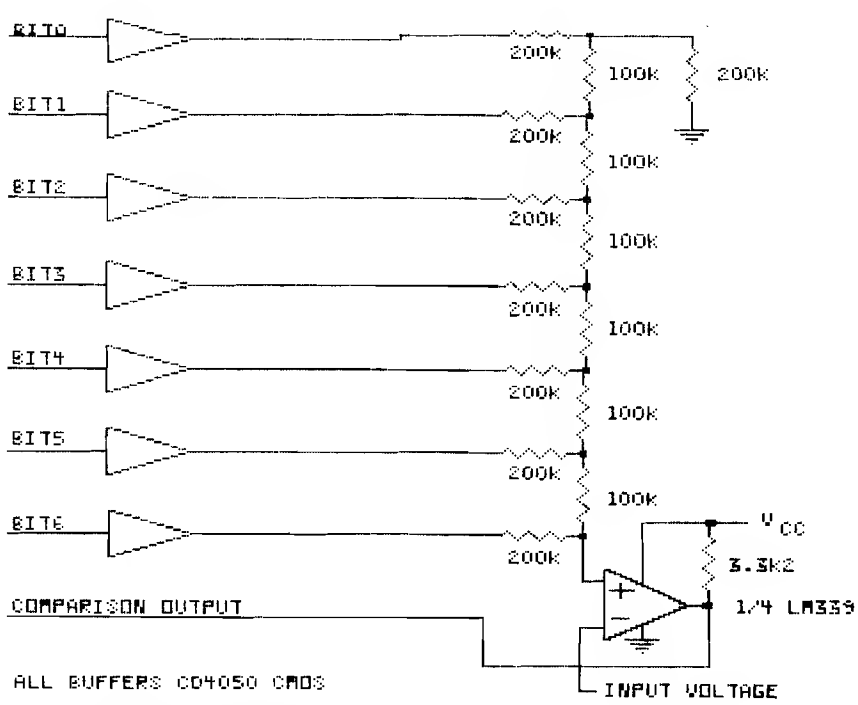


Figure 2. Add CMOS buffers to TTL output ports to increase accuracy and allow adjustment of full scale output.

Another interface design uses the serial or printer port to drive the circuit shown in Figure 3. Commodore owners can use the RS-232 output directly, since it comes out at TTL levels. Other computers probably will have standard RS-232 levels, but the transistors shown will shift levels satisfactorily. In this design, binary codes are output by the counter. If there is no activity on the channel for a while, the reset network resets the counter to zero count. The time constant of this circuit is long enough that normal clocking activity will keep the reset off. When you want to make a conversion, read the Serial In line to be sure that the comparator output is low. Clock the counter one count at a time, watching the Serial In line for a transition while counting the clock pulses internally. When the transition happens, subtract one count to get the binary value of the voltage.

Owners of the Radio Shack Color Computer have the equivalent of four A/D converters built in - the joystick ports. This A/D converter is very similar in principle to the circuit of Figure 1, as you can determine if you have the Color Computer Service Manual. Note that these ports have only six bit resolution - one part in 64 - and the accuracy may not be adequate for your needs. If you want to practice A/D techniques, the joystick ports can be useful. The input to the joystick ports are made on pins 1 and 2 of each joystick connector. Follow the instructions in the Color Computer manuals for reading the joystick positions.

There are a number of different types of A/D converter technologies available. The basic concept of A/D conversion is that the voltage to be measured is compared to a standardized voltage and the result expressed as a digital value. However, it is also possible for input voltage (or current) to be converted to a frequency proportional to the voltage. ICs which implement frequency/voltage (F/V) conversion are inexpensive and easy to use; we will do some experiments with them later.

A converter such as is shown in Figure 3 is a linear or serial converter. To make a conversion we must output each binary value in turn until the unknown voltage is reached. If you

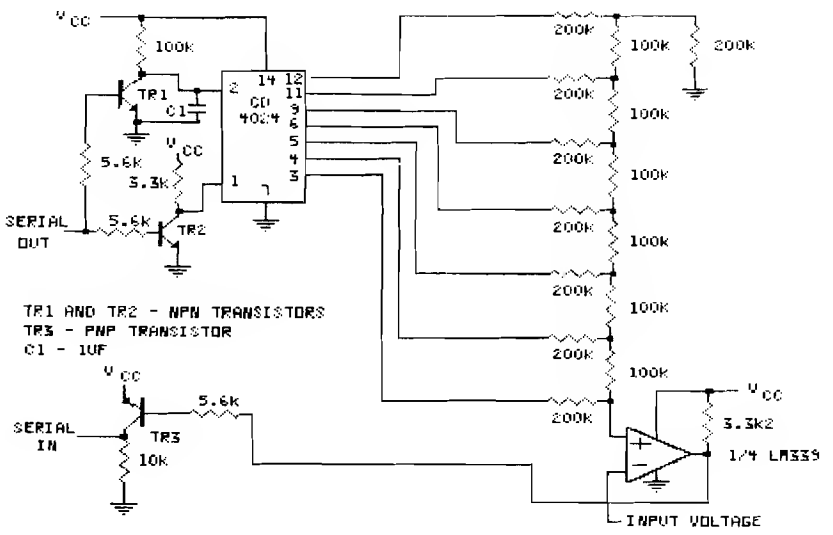


Figure 3. The A/D converter of Figure 1 adapted to operation on the Serial port.

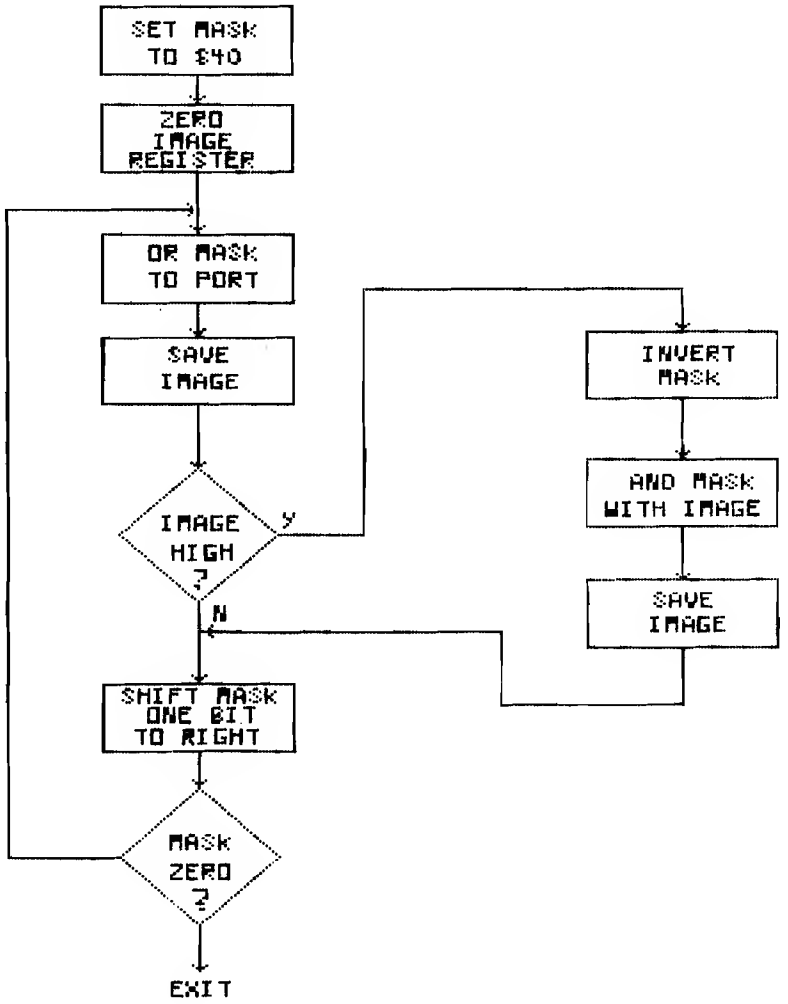


Figure 4. A flowchart implementing a successive approximation algortihm.

look at the output of Figure 3 on an oscilloscope, you will see a linear stair-step voltage with each step equal to .0391 volts for 5 volts full scale.

The converters of Figures 1 and 2 can be driven with a *successive approximation* algorithm. This method sets the MSB of the converter and checks to see if half-scale output is greater than the unknown voltage. If so, the MSB is turned off and each successive lower bit tried. If the result is lower than the unknown, that bit is left set, and the process is repeated with the remaining bits. The final answer is the binary representation of the voltage, and the answer will be less than one bit lower than the unknown voltage.

Figure 4 is a flowchart implementing the successive approximation algorithm for 7-bit converters such as Figure 1. The basic flow is to output each bit, beginning with the MSB (bit mask starts with $40). In case you have forgotten, individual bits are turned on in a PIA by a logical OR between the port image and the current mask bit (third block). If the bit being tested is too high, the bit is reset using a logical AND of the port image and the inverted bit mask. A new bit mask is created for each test cycle by shifting the starting mask right.

The following mini-program, written in Color Computer BASIC, illustrates the successive approximation algorithm by printing the port value which would be output after each test.

```
100 MA=50:IM=0
150 IM=MA OR IM
160 INPUT "H OR L?";A$
170 IF A$="H" THEN 500
180 PRINT IM
190 IF MA=0 THEN 600
195 GOTO 150
500 MI=255-MA
510 IM=IM AND MI
520 GOTO 180
600 PRINT "CONVERSION
    COMPLETE":STOP
```

This BASIC program closely parallels the flowchart of Figure 4 except that BASIC automatically saves the variables. Line 170 allows you to verify that the correct bit pattern will be output; in effect, you specify if the test will be 'high' or 'low' by entering **H** or **L** from the keyboard.

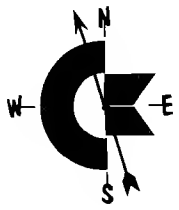
There is one loose end to our discussion - how accurate are the converters we have shown? The R-2R ladder's accuracy depends upon having a close match between all the 100K resistors and 200K resistors, and a ratio between the two values very close to 2:1. You can use an ordinary ohmmeter (or an average digital ohmmeter) to sort a number of 100K resistors. Try to find six resistors exactly alike, rather than exactly 100K. Pick the 200K resistors to be equal to the measured resistance of two selected 100K resistors in series. Use as little soldering heat as possible while getting good connections.

The correct resistor values will create repeatable step sizes, and full scale accuracy will depend upon adjusting the Vcc to the PIA (Figure 1), CMOS buffers (Figure 2) or the counter (Figure 3). If you have a very accurate voltmeter for calibration, your A/D can be more accurate than you imagine!

MICRO™

MICRO™

Commodore Compass



by Loren Wright

Modems, Etc.

Modems

I've managed to accumulate a number of products for review, so I thought I would cover a few together under this title.

Recently, I took the plunge into telecommunications with the purchase of a VICMODEM (Model 1600), took out my review copy of SuperTerm, and proceeded to try to get on line with CompuServe. The first problem was that I couldn't use the VICMODEM with my phone. Not only does the phone have to be modular, but all the electronics must be in the base of the telephone. Slimline or Trimline phones do not work! I was able to get the VICMODEM to work with someone else's phone. The procedure is a little awkward: you have to make the call, listen for the high-pitched squeal, then remove the modular plug from the handset, and plug it into the modem. The VICMODEM plugs into the VIC or C-64's UserPort, and has exactly three external features: the modular phone-plug receptacle, an activity light, and an answer/originate switch.

The answer to my problem was to buy a Commodore AUTOMODEM (Model 1650). This was a little more expensive ($100 vs. $60), but well worth the difference. Not only does the AUTOMODEM work with my telephone, but it also has a number of other very useful features. The AUTOMODEM doesn't really even require that you have a telephone! There are two modular jacks on the side of the 1650. In one you plug a line from the walljack (included with the modem); in the other you plug the telephone. There's a switch on the modem between the telephone and the modem. Another switch controls the answer/originate mode, and a third switch controls full/half duplex. The reason the unit is called AUTOMODEM is that it can be programmed to automatically dial a number and automatically answer when you aren't around.

These two modems are not the only ones that work with the VIC and C-64, but they are the cheapest ready-to-use units. Both work only at 300 baud,so if you need faster speeds you will have to buy one of the more expensive units such as the Hayes SmartModem 1200.

Terminal Programs

Both the VICMODEM and the AUTOMODEM require software to do anything. There's a cassette included with each that has a VIC terminal program on one side, and a Commodore 64 program on the other. The C-64 AUTOMODEM program implements the auto-dial and auto-answer features.

These programs are rather minimal, however. They are sufficient to use CompuServe, Dow Jones, The Source, and most bulletin boards, but functions such as LOADing material from and SAVEing to DISK, printing documents, handling programs, and ASCII conversion, are difficult or impossible. For these tasks you require a terminal program.

Perhaps the best program on the market for the Commodore 64 or VIC is SuperTerm from Midwest Micro, Inc. (311 W. 72nd St., Kansas City, MO 64114). This program can handle just about any communications situation imaginable. The baud, bits, parity, stopbits, and duplex are all selectable. The screen can be set to operate with 40, 80, or 132 columns, and there is a convenient 'freeze' mode to aid in reading the wider screen formats. The buffer can be edited before transmission or after it is received. Any of 42 different terminals can be emulated. Files can be converted from CBM to ASCII and vice versa. I haven't begun to test all the different possibilities.

SuperTerm is notable in that it supports printer streaming. That is, you can set it up so that everything that comes across the line goes directly to the printer. This requires Midwest Micro's Smart ASCII printer interface ($59.95). Smart ASCII works through the expansion port, not the serial port used by most other interfaces. Without this interface you must use the print-buffer procedure.

The only complaint I have with SuperTerm is the manual. Included with my review copy is Revision 1.1 of what they call "Preliminary Documentation." I do hope that the final documentation explains things a little more clearly and gives the user step-by-step instructions for such common operations as sending/receiving CBM programs, WordPro files, and UP/DOWN loading from CompuServe.

The $149.95 price is a little high, but certainly worth it if you want to be able to do practically anything with your modem. The VIC-20 version (not reviewed) requires a 16K expansion.

Those with less ambitious communications plans and/or smaller budgets may wish to consider Midwest Micro's other terminal programs. Also, I am told that there is a very powerful public domain 'Terminal 64' program available from Midnight Software (1238 Richland Ave., Lincoln, IL 62656) for a $15 media and handling fee. (Don't send disk and mailer -- just money! Ask for 'The Terminal Disk').

On-line Information Sources

Commodore includes information on

CompuServe and Dow Jones with both the VICMODEM and AUTOMODEM. The CompuServe information includes a user ID and password to let you get on the system right away for a free hour. Dow Jones also gives you a free hour, but you have to send away for the password and ID number. So far, I have spent a couple hours on CompuServe and have not really figured everything out yet. Commodore has decided to support CompuServe not only by including the information and free hour, but also by running a Commodore bulletin board service. Through this bulletin board (accessed from CompuServe by 'GO CMB') you can exchange messages, read Commodore bulletins and product announcements, download programs from their magazine articles, and participate in on-line conferences. You can leave messages for Commodore personnel to get hardware/software help, and most likely get an answer within a few days.

CompuServe is accessed through one of many access numbers throughout the US and Canada. If you are fortunate, as I was, there will be one that is a local call for you. In addition, you can use TYMNET, DATAPAC, and TELENET numbers, but using these costs extra. Once you have used up your free hour, it costs $6/hour that you are connected. This applies to evenings, nights, and weekends at 300 baud. Daytimes and 1200 baud service are much more expensive. There is no initiation fee, but some services, such as up-to-the-minute stock quotes cost extra.

Dow Jones News/Retrieval gives you access to Wall Street Journal news and other business information. The membership fee is waived for AUTOMODEM and VICMODEM purchasers. I have not yet tried Dow Jones.

The Source is another information source available. The hourly connect rate for evenings, nights and weekends is $7.50, and other rates are more expensive than CompuServe, too. There are account maintenance and minimum monthly fees as well. This means that you get a bill for $10 every month, even if you don't use the service. The biggest obstacle to the home user is the $100 membership fee. I was not able to try out The Source due to these financial impediments. It seems that The Source is geared to the business user, rather than the home user, and the presence of the Commodore bulletin board on CompuServe should swing you in favor of them anyway.

There's a great number of local bulletin boards springing up, and many of them have Commodore special interest groups (or SIG's, for short). Although these don't have the variety of services and information that CompuServe boasts, the cost is often no more than a local phone call. Using a bulletin board is a good way to hook up with other Commodore users. Ask a local computer store about bulletin boards in your area.

There are also more nationally oriented bulletin boards which are also free, except for the long distance call. These include Steve Punter's bulletin board, The Toronto PET User's Group, The Midnight/Gazette, and a great number of others. In a future column, I will publish a list of bulletin boards. Bulletin board operators send in your numbers, hours, specialties, etc., to get included in this list.

Loren Wright may be reached at 37 Adam Drive, Hudson, NH 03051, or as CompuServe 70626.636.

MICRO Program Listing Conventions

Commodore

LISTING	C64 KEYBOARD
Commands	
{CLEAR}	^ CLR
{HOME}	HOME
{INSERT}	^ INST
{DOWN}	CRSR DOWN
{UP}	^ CRSR UP
{RIGHT}	CRSR RIGHT
{LEFT}	^ CRSR LEFT
Colors	
{BLACK}	CTRL 1 BLK
{WHITE}	CTRL 2 WHT
{RED}	CTRL 3 RED
{CYN}	CTRL 4 CYN
{PURPLE}	CTRL 5 PUR
{GREEN}	CTRL 6 GRN
{BLUE}	CTRL 7 BLU
{YELLOW}	CTRL 8 YEL
{RVS}	CTRL 9 RVS ON
{RVSOFF}	CTRL 0 RVS OFF
{ORANGE}	= 1
{BROWN}	= 2
{GREY 1}	= 3
{GREY 1}	= 4
{GREY 2}	= 5
{LT GREEN}	= 6
{LT BLUE}	= 7
{GREY 3}	= 8
Functions	
{F1}	f1
{F2}	^ f2
{F3}	f3
{F4}	^ f4
{F5}	f5
{F6}	^ f6
{F7}	f7
{F8}	^ f8
Special Characters	
{PI}	^ Pi Char
{POUND}	Pound Sign
{UP ARROW}	Up Arrow
{BACK ARROW}	Back Arrow

Atari

Conventions used in ATARI Listings.

Normal Alphanumeric appear as UPPER CASE:
SAMPLE
Reversed Alphanumeric appear as lower case:
yES (y is reversed)
Special Control Characters in quotes appear as:
{command} as follows:

Listing	Command	ATARI Keys
{UP}	Cursor Up	ESC/CTRL -
{DOWN}	Cursor Down	ESC/CTRL =
{LEFT}	Cursor Left	ESC/CTRL +
{RIGHT}	Cursor Right	ESC/CTRL *
{CLEAR}	Clear Screen	ESC/CLEAR
{BACK}	Back Space	ESC/BACK S
{TAB}	Cursor to Tab	ESC/TAB
{DELETE LINE}	Delete Line	ESC/SHIFT DELETE
{INSERT LINE}	Insert Line	ESC/SHIFT INSERT
{CLEAR TAB}	Clear Tab Stop	ESC/CTRL TAB
{SET TAB}	Set Tab Stop	ESC/SHIFT TAB
{BEEP}	Beep Speaker	ESC/CTRL 2
{DELETE}	Delete Char.	ESC/CTRL BACK S
{INSERT}	Insert Char.	ESC/CTRL INSERT
{CTRL A}	Graphic Char. where A is any Graphic Letter Key	CTRL A

Non-Keyboard Commands

{DIS=}	CHR$(8)
{ENB=}	CHR$(9)
{LOWER CASE}	CHR$(14)
{UPPER CASE}	CHR$(142)
{^RETURN}	CHR$(142)
{DEL}	CHR$(20)
{SPACE}	CHR$(160)

Notes:

1. ^ represents SHIFT KEY
2. = represents Commodore Key in lower left corner of keyboard
3. CTRL represents CTRL Key
4. Graphics characters represented in Listing by keystrokes required to generate the character
5. A number directly after a {SYMBOL} indicates multiples of the SYMBOL: {DOWN6} would mean DOWN 6 times

Coming Attractions ...

MARCH features mathematical techniques:

- Least Squares Curve Fitter
- Plotting Fractals
- Alpha Beta Technique
- Plotting Bezier Curves

DOSPLUS Part 3 will contain:

- Machine Language Monitor
- Printer Formatting Program
- Repeat Key Toggle

A **Credit Register** program will keep track of up to eight separate credit card accounts.

APRIL begins a two-part update for *What's Where in the Apple*. The update contains the information required to use this classic work with the Apple IIe. The complete update will include fourteen pages of detailed memory maps for the Apple IIe.

DOSPLUS Part 4 provides techniques and programs to allow BASIC programs to be hidden under ROM and swapped with the currently active BASIC program, plus a time and alarm routine.

The MICRO Cover

This month's cover photo by **Thaworn Phatinawin** of Long Beach, California is an entry in the MICRO Graphics Contest. It was generated on a Commodore 64 using the Koala Pad graphics package.

Advertiser's Index

A B Computers	4
Amplify	32
Artemis Systems	52
CompuTech	56
Computer Mail Order	44,45
Computer Software Associates	56
Datamost, Inc	20,21
Eastern House Software	1
Excert, Inc.	54
Hollywood Hardware	50
Jack Leverenz Sailing School	17
J & M Software	2
Kiwi Software	52
Leading Edge	Back Cover
MICRO Magazine	48,63
Micro Spec	64
Micro Ware	37
Midwest Micro Inc	38
Nibble	29
Percom Data	Inside Front Cover
Performance Micro Products	32
Perry Peripherals	58
Protecto Enterprizes	11, 12, 13
RH Electronics, Inc	22
S G C	7
Safeware	18
Skyles Electric Works	46
Star Micronics	Inside Back Cover
Strom Systems Inc	62
Titan Technogies, Inc	43
Totl Software	8
Winders & Geist, Inc	3
Zanim Systems	60

National Advertising Representatives

Home Office:
Cindy Kocher, Advertising Manager
P.O. Box 6502
Chelmsford, MA 01824 (617) 256-3649

Mid-West Territory:
Thomas Knorr & Associates
Thomas H. Knorr, Jr.
333 N. Michigan Avenue, Suite 401
Chicago, Illinois 60601 (312) 726-2633

serving: Ohio, Oklahoma, Arkansas, Texas, North Dakota, South Dakota, Nebraska, Kansas, Missouri, Indiana, Illinois, Iowa, Michigan, Wisconsin, and Minnesota.

West Coast:
The R.W. Walker Co., Inc.
Gordon Carnie
2716 Ocean Park Boulevard, Suite 1010,
Santa Monica, California 90405 (213) 450-9001

serving: Washington, Oregon, Idaho, Montana, Wyoming, Colorado, New Mexico, Arizona, Utah, Nevada, California, Alaska, and Hawaii (also British Columbia and Alberta, Canada)

Be confident because your PowerType Daisywheel correspondence becomes you.

When you can't personally be there, only the clearest typed correspondence should be your substitute. With a PowerType Daisywheel printer your documents look highly professional. And so do you!

PowerType. It's "typewriter friendly." Using a simple drop-in ribbon cassette, it bi-directionally types executive quality correspondence at 18 cps with a print wheel that holds 96 flawless characters.

Designed for personal or business applications, PowerType's carriage accepts paper that ranges from letter to legal size, from fanfold to roll to cut sheet. You can set right and left margins, vertical and horizontal tabs.

Plus, of course, PowerType has both serial and parallel interfaces to enable it to connect to just about any personal or business computer.

So the next time you're going face to face through the mail, rely on PowerType. It will help you make a professional impression. And that's always very becoming.

star™
micronics • inc

THE POWER BEHIND THE PRINTED WORD.

Computer Peripherals Division
P.O. Box 612186, Dallas/Ft. Worth Airport, TX 75261 (214) 456-0052

REMEMBER.

ELEPHANT™ NEVER FORGETS.

A full line of top-quality floppies, in virtually every 5¼" and 8" model, for compatibility with virtually every computer on the market. Guaranteed to meet or exceed every industry standard, certified 100% error-free and problem-free, and to maintain its quality for at least 12 million passes (or over a lifetime of heavy-duty use).

Contact Dennison Computer Supplies, Inc., 55 Providence Highway, Norwood, MA 02062 or call toll-free 1-800-343-8413. In Massachusetts, call collect (617) 769-8150. Telex 951-624.

Dennison